Impossibility of Indifferentiable Iterated Blockciphers from 3 or Less Primitive Calls*

Chun Guo[1 2 3](✉), Lei Wang[4](✉), and Dongdai Lin[5](✉)

[1] School of Cyber Science and Technology, Shandong University, Qingdao, China
[2] Key Laboratory of Cryptologic Technology and Information Security of Ministry of Education, Shandong University, Qingdao, Shandong, 266237, China
[3] Shandong Research Institute of Industrial Technology, Jinan, Shandong, 250102, China
[4] Shanghai Jiaotong University, Shanghai, China
[5] Institute of Information Engineering, Chinese Academy of Sciences, China
chun.guo.sc@gmail.com,wanglei_hb@sjtu.edu.cn,ddlin@iie.ac.cn

Abstract. Virtually all modern blockciphers are *iterated*. In this paper, we ask: to construct a secure iterated blockcipher "non-trivially", how many calls to random functions and permutations are necessary?
When security means *indistinguishability from a random permutation*, optimality is achieved by the Even-Mansour scheme using 1 call to a public permutation. We seek for the arguably strongest security *indifferentiability from an ideal cipher*, a notion introduced by Maurer et al. (TCC 2004) and popularized by Coron et al. (JoC, 2014).
We provide the first generic negative result/lower bounds: when the key is not too short, no iterated blockcipher making 3 calls is (statistically) indifferentiable. This proves optimality for a 4-call positive result of Guo et al. (Eprint 2016). Furthermore, using 1 or 2 calls, even indifferentiable iterated blockciphers with polynomial keyspace are impossible.
To prove this, we develop an abstraction of idealized iterated blockciphers and establish various basic properties, and apply Extremal Graph Theory results to prove the existence of certain (generalized) non-random properties such as the boomerang and yoyo.

1 Introduction

Iterated blockciphers. Virtually all modern blockciphers, e.g., DES, AES, PRESENT, Skinny, are designed via iteration [2]. These even include theoretical constructions such as the Luby-Rackoff [40], Iterated Even-Mansour (IEM) ciphers [23,11,1,30] and others [21,29]. In fact, the initialization algorithms of some stream ciphers [50] also follow the iteration paradigm.

The idea of iteration dates back to Shannon [47] or even earlier practice of product ciphers. In general, an iterated structure creates a (usually weak) keyed permutation, typically called its *round*, in a "non-trivial" manner, and then composes such rounds till gaining enough security. By "non-trivial", the round has to employ smart ideas to resolve non-invertibility of functions [40] or combine

* (✉): Corresponding authors. Full version available at [28].

keys with keyless permutations [23,11,1]. Such constructs also constitute natural transformations between (pseudo)random functions and (pseudo)random permutations [40,23,1,16,21], which are fundamental in modern cryptography.

While provably secure blockciphers remain out of reach, there is a definite belief that with sufficient iterations, the iterated paradigm does yield enough security. The primary security notion for a blockcipher is *indistinguishability from a random permutation*, i.e., no adversary with bounded oracle queries and black-box access to a permutation can distinguish whether it is interacting with the blockcipher under a random key or a perfectly random permutation. This has probably been the most widely used security assumption for blockciphers. In fact, with certain idealized assumptions and sufficient iterations, the aforementioned Luby-Rackoff [40], IEM [23,11,30] and Swap-Or-Not [29] have been proven indistinguishable (and bounds usually increase with rounds).

The ideal cipher model. Albeit the de-facto standard, indistinguishability is insufficient for a number of important blockcipher-based cryptosystems. For example, some real-world protocols such as f8 and f9 [33] crucially rely on the stronger related-key security of blockciphers [6]. Even worse, in blockcipher-based hash functions [10,9], the adversary can control both the message and the key of the blockcipher and exploit "known-key" or "chosen-key" attacks [37,8] to break collision- or preimage-resistance of the hash. In fact, a mere PRP cannot yield black-box construction of collision resistant hash [48].

Hence, cryptographers have modeled a reliable (κ, n)-blockcipher (i.e., a blockcipher with κ-bit keys and n-bit blocks) as an *ideal cipher* (IC), i.e., a family of 2^κ independent n-bit random permutations that is public to all entities. This is known as the *ideal cipher model* (ICM), and it turned out crucial for proving security for blockcipher-based schemes when the PRP assumption is not enough [9,10,35]. While remaining a heuristic approach [12,41], a proof in the ideal cipher model is typically considered a good indication of security from the point of view of practice. Meanwhile, "being close to ideal" becomes a new standard for blockcipher design and evaluations—much like "being close to a random oracle" for hash functions [15,7]. In fact, distinguishing blockcipher algorithms from "ideal" has been recognized as an important attack vector [37,8].

Indifferentiability. While ICs are unachievable in the standard model [12,41], it remains an interesting problem to build ICs from other public ideal functions. This class of problem shall be addressed with *indifferentiability* introduced by Maurer et al. [41] and popularized by Coron et al. [15]. Indifferentiability is a simulation-based framework that helps assess whether a construction of a target primitive $A^{\mathbf{B}}$ from a lower-level ideal primitive $\mathbf{B}$ is "structurally close" to $\mathbf{C}$, the ideal version of $A^{\mathbf{B}}$ (e.g., the case where A is the IEM cipher, $\mathbf{B}$ is the random permutation $\mathbf{P}$ and $\mathbf{C}$ is an IC was considered in [1]). $A^{\mathbf{B}}$ is *indifferentiable from* $\mathbf{C}$, if for any *differentiator* D there exists an *efficient simulator* $S^{\mathbf{C}}$ querying $\mathbf{B}$ such that the two systems $(A^{\mathbf{B}}, \mathbf{B})$ and $(\mathbf{C}, S^{\mathbf{C}})$ are indistinguishable in the view of D. Indifferentiability comes equipped with a composition theorem [41] which implies that a large class of protocols (see [43,20] for restrictions) are provably secure in the ideal-**B** model if and only if they are provably secure in the ideal-**C** model. Since stronger notions are unachievable in general [43,20], indifferentiability is arguably the strongest security notion for cryptosystems. Due to this and due to the importance of composition, indifferentiability has been applied to various cryptosystems, including iterated hash [15,7], blockciphers [16,1,21], authenticated encryption [5] and public-key schemes [51].

Therefore, it has been an important direction to evaluate indifferentiability of popular blockcipher constructions [16,1]. The first feasibility was the key-prepended Feistel cipher of Coron et al. [16], which iterates $\Psi^{\mathbf{F}}(K, x_L \| x_R) := x_R \oplus \mathbf{F}(K \| x_L) \| x_L$ with $x_L, x_R \in \{0,1\}^{n/2}$ and $\mathbf{F}$ a public random function. Coron et al. proved indifferentiability with 14 rounds [16] and established equivalence of ideal models. This was later improved to 10 [17] and 8 rounds [19].

Another line of work established indifferentiability for the mentioned IEM ciphers. Concretely, a t-round IEM cipher employs t n-bit random permutations $\mathbf{P}_1, \ldots, \mathbf{P}_t$ and $t+1$ key derivation functions $\mathsf{kd}_0, ..., \mathsf{kd}_t : \{0,1\}^\kappa \to \{0,1\}^n$, and is defined by iterating $\mathrm{EM}_\ell^{\mathbf{P}}(K, x) := \mathsf{kd}_\ell(K) \oplus \mathbf{P}_\ell(\mathsf{kd}_{\ell-1}(K) \oplus x)$. When $\mathsf{kd}_0 = ... = \mathsf{kd}_t = \mathbf{F}$ for a random function $\mathbf{F} : \{0,1\}^\kappa \to \{0,1\}^n$, positive results were first proven at 5 rounds [1] and later tightened to 3 rounds [27]. When $\mathsf{kd}_0, ..., \mathsf{kd}_t$ are the identity function id, positive results were first proven at 12 rounds [39] and later tightened to 5 rounds [18].

Lower bounds? We seek for understanding the *complexity* and ask: to have a "non-trivial", provably secure iterated (κ, n)-blockcipher, how many calls to the primitives are *necessary*? Such results may shed lights on *limits on efficiency of widely used paradigms* as well as *boundary of blockcipher designs*.

By "non-trivial", we mean the construction must use some ideas. E.g., if an oracle $\mathcal{O}$ already contains an exponential number of independent n-bit random permutations, then $E^{\mathcal{O}}$ can trivially instantiate an indifferentiable blockcipher. With this in mind, we introduce an oracle $\mathcal{P}$ that "provides all but the goal".

In detail, $\mathcal{P} = (\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_{|\mathcal{I}|})$ is a family of independent random permutations indexed by $i \in \mathcal{I}$, where $\mathbf{P}_i : \{0,1\}^{m(i)} \to \{0,1\}^{m(i)}$ for an integer function $m : \mathcal{I} \to \mathrm{poly}(n)$. The set $\mathcal{I}$ is partitioned into $\mathcal{I}_{\le n}$ and $\mathcal{I}_{>n}$, such that $i \in \mathcal{I}_{\le n}$ if and only if $m(i) \le n$. To avoid trivial results, we require $|\mathcal{I}_{\le n}| = O(\mathrm{poly}(n))$, so that $\mathcal{P}$ cannot offer exponentially many n-bit permutations. For $i \in \mathcal{I}_{>n}$, it can be $m(i) \gg n$, and an indifferentiable random function/injection can be built by calling such a wide permutation once [14,5]. Thus, such an oracle $\mathcal{P}$ essentially offers the "maximal" power to the constructions. As will be detailed in Sect. 5.1, existing constructions [16,1,39] can be seen as defined upon $\mathcal{P}$.

The status, of course, depends on the security notion. W.r.t. *indistinguishability*, a single permutation-call is already sufficient using the Even-Mansour scheme [23]. We seek for bounds w.r.t. *indifferentiability*. *Specific lower bounds* have been shown: Feistel ciphers [16,19] consume *at least* 6 random function calls, while IEM ciphers need 4 random function/permutation calls [1,18,27]. Despite this and the fruitful positive results mentioned before, no *general lower bounds* are publicly known (except that a polynomial-length random string is insufficient [41]) due to its challenging nature: the adversarial goal is not as clear as [9,44,3] (which simply finds collisions or pre-images), and one has to pinpoint "non-random" properties that are exploitable within polynomial-queries (unlike [44,3]) in various cases, and further prove that interactions with ideal ciphers and *all possible simulators* are unlikely to admit such properties.

Our results. We prove the first general lower bound: *no iterated blockcipher making 3 or less calls to the oracle $\mathcal{P}$ is statistically indifferentiable from ideal ciphers.* This proves *optimality* for the mentioned 4-call positive result [27].

Model and settings. We consider *iterated blockciphers* that can be written as the composition of *rounds* using keys or derived subkeys. Every *round* is essentially a simpler "1-call" blockcipher *making exactly 1 call to* $\mathcal{P}$, and the total number of $\mathcal{P}$-calls made by the rounds and the key derivation function is a constant.

More concretely, to model rounds/1-call ciphers, we define $E1^{\mathcal{P}}(K,x) := \varphi^{out}\big(K, \mathcal{P}(\varphi^{in}(K,x)), x\big)$ with keyspace $\mathcal{K}$ and domain $\{0,1\}^n$. The *input function* φ^{in} maps $(K,x) \in \mathcal{K} \times \{0,1\}^n$ into a query (i,δ,z) to $\mathcal{P}$, where $\delta \in \{+,-\}$ indicates the direction, $i \in \mathcal{I}$ indexes the queried permutation and $z \in \{0,1\}^{m(i)}$ is the concrete query. The *output function* φ^{out} maps the key K, the $\mathcal{P}$ response $z' = \mathcal{P}(\varphi^{in}(K,x))$ and the plaintext x to the ciphertext y.

$E1^{\mathcal{P}}$ must admit efficient inversion within 1 $\mathcal{P}$-call as well. Thus, it is defined $(E1^{-1})^{\mathcal{P}}(K,y) := \gamma^{out}\big(K, \mathcal{P}(\gamma^{in}(K,y)), y\big)$ for two other input and output functions γ^{in} and γ^{out}. Arguably, this covers all blockciphers using a single oracle call (which resembles [9]). See Fig. 1 for illustration.

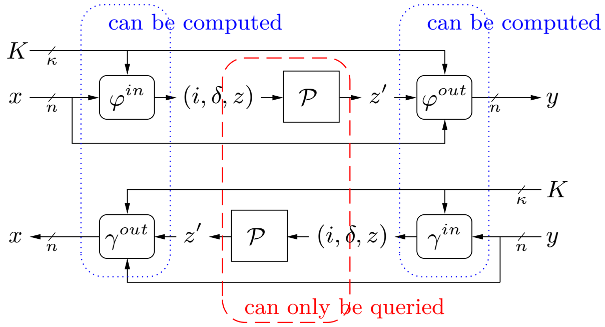

Fig. 1. The general blockcipher $E1^{\mathcal{P}}$ making a single call to its oracle $\mathcal{P}$ for enciphering (up) and deciphering (bottom). $\varphi^{in}, \varphi^{out}, \gamma^{in}$ and γ^{out} are arbitrary (e.g., can be highly non-linear) deterministic and oracle-independent functions, and are computable by the differentiator (as indicated). $\mathcal{P} = (\mathbf{P}_1, \mathbf{P}_2, ...)$ is the mentioned family of random permutations, and it only offers oracle access to the differentiator.

Then, for our model of a t-call iterated blockcipher $Et^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$, the keyspace $\mathcal{K}$ is partitioned into disjoint sets $\mathcal{K}^{(0)}, \mathcal{K}^{(1)}, ...\mathcal{K}^{(t-1)}$, such that for all $K \in \mathcal{K}^{(\ell)}$, it has

$$Et^{\mathcal{P}}(K,x) = \Pi^{\mathcal{P}}_{j_{\ell,t-\ell}}\Big(K\|s, ...\Pi^{\mathcal{P}}_{j_{\ell,2}}\big(K\|s, \Pi^{\mathcal{P}}_{j_{\ell,1}}(K\|s,x)\big)...\Big), \qquad (1)$$

where:

(i) $s = \mathsf{kd}^{\mathcal{P}}(K)$ is a subkey and $\mathsf{kd}^{\mathcal{P}}$ makes ℓ calls to $\mathcal{P}$, and

(ii) For each $\ell \in \{0, ..., t-1\}$ and each $\alpha \in \{1, ..., t-\ell\}$, $j_{\ell,\alpha} = \ell(\ell-1)/2 + \alpha$ (so that $Et^{\mathcal{P}}$ is defined upon $\ell(\ell+1)/2$ distinct rounds $\Pi_1, ..., \Pi_{\ell(\ell+1)/2}$), and the round $\Pi^{\mathcal{P}}_{j_{\ell,\alpha}}$ is a 1-call cipher.

See Fig. 2 for illustration of $E2^{\mathcal{P}}$ and Figs. 11 and 12 for pseudocode of $E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$. This unifies virtually all existing blockcipher constructions. While the same oracle $\mathcal{P}$ is used everywhere in $Et^{\mathcal{P}}$, our subsequent attacks never utilize this oracle reusing, and are applicable even if multiple $\mathcal{P}_1, \mathcal{P}_2, ...$ are used.

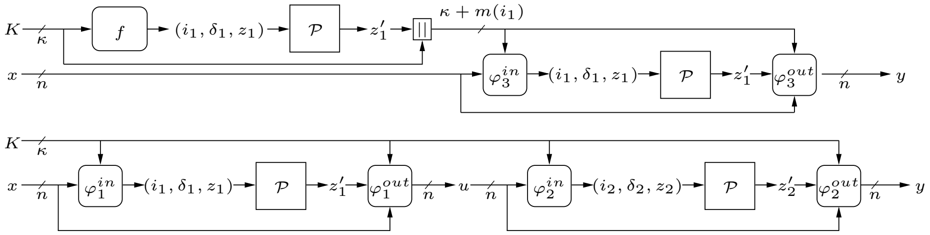

Fig. 2. Encipherment of 2-call iterated blockciphers. (Top) Using one $\mathcal{P}$-call for a key derivation $\mathsf{kd}^{\mathcal{P}}(K) = \mathcal{P}(f(K))$. The function f is deterministic and oracle-independent; (Bottom) Using two $\mathcal{P}$-calls for two rounds, without idealized key derivations.

Our reasoning relies on four *Fundamental Properties* that stem from *the notions of blockciphers* and *of t-call oracle procedures*. Namely, a blockcipher oracle procedure $E^{\mathcal{P}}$ should be **efficiently invertible**, **deterministic**, and enjoy an **oracle-independent** description. Moreover, it should be **non-degenerate** (i.e., $E^{\mathcal{P}}$ cannot be "simplified" in terms of $\mathcal{P}$ calls). We refer to Sect. 3.1 or 4 for details. Our setting may find broader applications in symmetric cryptography. As a side remark, our crucial use of invertibility solves an open problem of [5].

Differentiability of E1, E2 and E3. With the above models, we prove our main result by characterizing $E1^{\mathcal{P}}$ and extending to $E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$.

In detail, for 1-call ciphers $E1^{\mathcal{P}}$, we fully characterize its properties, solely based on the *Fundamental Properties*. In summary, as long as the keyspace has $|\mathcal{K}| \geq 2|\mathcal{I}_{\leq n}| + 1 = O(\text{poly}(n))$ (thus, even polynomial keyspace is unachievable!),[6] we can find either $\Omega(\text{poly}(n))$ "inverse-free" encipherments that collide on the $\mathcal{P}$-call and use an entropy-based differentiating approach [41], or find two "non-inverse-free" encipherments $E1^{\mathcal{P}}(K, x)$ and $E1^{\mathcal{P}}(K', x')$ with $\varphi^{in}(K, x) = \varphi^{in}(K', x')$ and use a special regularity property of φ^{in} and γ^{in} to distinguish. We refer to Sect. 3.2 or Theorem 1 for details.

For 2-call iterated cipher $E2^{\mathcal{P}}$, if $\mathcal{K}^{(1)}$ is large enough, i.e., $E2^{\mathcal{P}}$ invokes key derivation for sufficiently many keys, then our differentiators derives $O(\text{poly}(n))$ keys to "collapse" the cipher to a 1-call instance, which has been attacked. On the

[6] Though, trivial constructions with $|\mathcal{K}| \leq |\mathcal{I}_{\leq n}|$ exist since $\mathcal{P}$ may offer $|\mathcal{I}_{\leq n}|$ independent n-bit RPs.

other hand, if $\mathcal{K}^{(0)}$ dominates, i.e., $E2^{\mathcal{P}}$ is a general 2-round blockcipher for most keys, then as long as $\mathcal{K}^{(0)}$ is large enough $|\mathcal{K}^{(0)}| \geq \big(6\big(3|\mathcal{I}_{\leq n}|\big)^{\frac{1}{n}} + 5\big)|\mathcal{I}_{\leq n}| + 1 = O(\mathrm{poly}(n))$, we can exhibit a general yoyo distinguisher and breaks its *correlation intractability* (a weaker security notion than indifferentiability). We refer to Sect. 3.3 or Theorem 2 for details.

For 3-call iterated ciphers $E3^{\mathcal{P}}$, if $\mathcal{K}^{(2)}$ or $\mathcal{K}^{(1)}$ is large enough then we again "collapse" it to 1- or 2-call ciphers by deriving $\mathrm{poly}(n)$ keys. If $E3^{\mathcal{P}}$ is a general 3-round cipher for most keys $\mathcal{K}^{(0)} \subseteq \{0,1\}^{\kappa}$ with $\kappa \geq 2m_{max} \log_2 |\mathcal{I}_{\leq n}| + 2m_{max}n + 6m_{max} + 4 = \Theta(\mathrm{poly}(n))$, $m_{max} := \max_{i \in \mathcal{I}} m(i)$, we exhibit a universal differentiator that (interestingly) has attack advantage either at least $1/\mathrm{poly}(n) - \mathrm{negl}(n)$ or at least $1 - \mathrm{negl}(n)$, where the concrete polynomial and negligible functions depend on the input functions in the three rounds. We refer to Sect. 3.4 or Theorem 3 for details.

A crucial step is to show the existence of certain non-random properties, which is non-obvious in the general 2- and 3-call ciphers. To this end, we apply Extremal Graph Theory [32,38,24], which bound the maximal number of edges in (bipartite) graphs that do not contain certain structures (a.k.a. *Zarankiewicz numbers* [24]). We refer to Sect. 3 for more detailed overview.

Discussion: blockcipher designs. A recent trend is to revisit blockcipher structures and squeeze efficiency for MPC and ZKP settings: see [26] and the references therein. We hope that our work could be a step towards unifying relevant theoretical discussions and shed lights on the "boundary" of designs. We summarize some of our conclusions as follows.

(i) *Expense of overcoming non-invertibility*: if a round/1-call cipher $E1^{\mathcal{P}}(K, x)$ want to be inverse-free for some (K, x) (e.g., when using non-invertible primitives), then $E1^{\mathcal{P}}(K, \cdot)$ must admit severe weakness, regardless of its design.

(ii) *Unhelpfulness of wide permutations*: wide permutations with width$> n$ are not "more helpful" in constructing n-bit blockciphers, even if exponentially many are available. This might be another explanation on the difficulty in designing format-preserving encryption schemes (see e.g., [22]).

(iii) *Optimality of popular structures (e.g., the IEM ciphers [18,27])*, in the sense that *no other choice can be better*. This provides the first "excluding-type" theoretical support for practical paradigms.

Besides, since an indifferentiable iterated cipher needs at least 4 calls, our result may be viewed as a theoretical evidence of the advantage (in terms of efficiency) of permutation-based cryptography. Though, we remark that the usual caveats regarding the ideal model apply to this paper: as we consider information-theoretic adversaries, our results do not imply security upper bounds on real-world, computationally bounded adversaries.

Lower bounds: functionality transformations vs. small-to-big. Cryptographic constructs consist of two categories: *functionality transformations* and *small-to-big transformations*. The former achieves "non-trivial" new functionality (e.g., our case), while the latter achieve domain or range extension (e.g., PRGs extend range, while hashes extend domain).

A number of existing efficiency lower bounds concerned with *small-to-big transformations*, including hash functions [36,25,9,44,5], PRGs [25,31], signatures [25,3], encryption [25] and injections [5]. A core idea typically employed by these proofs is to apply the pigeonhole principle to force the scheme making the same sequence of primitive calls for exponentially many inputs. This results in either attacks [9,44,5] or unconditional cryptography [36,25,3].

Despite exciting black-box separations [46,34,4], efficiency lower bounds on *functionality transformations* are relatively rare. Our problem *is functionality transforming*: we allow to use wide permutations on $\geq \kappa + n \gg n$ bits, and the domain of our target $E : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ is thus *not larger*. This difference is crucial, as pigeonhole principle cannot ensure collisions and we have to rely on other properties such as non-degeneracy (see Lemma 3).

Notably, with our oracle $\mathcal{P}$, relevant impossibility results become possible:

(i) A compression function with enough collision or even indifferentiability security can be built using just 1 call to $\mathcal{P}$ via *truncation* [14];
(ii) An indifferentiable injection (or authenticated encryption) can be built using just 1 call to $\mathcal{P}$ via *Encode-then-Encipher* [5].

Still, indifferentiable iterated blockciphers *cannot* be built within 3 calls. These sharp contrasts emphasize the differences between our setting and [9,5].

In a more restricted setting termed *Linicrypt* [13], i.e., cryptosystems are built from random block functions and *linear* diffusion functions, impossibility results regarding encryption [42,13] and circuit garbling [13] exist.

Future directions. Indeed, blockciphers are *not* necessarily iterated: we serve examples in full version [28]. Intuitively, such designs are weaker than iterated ones with the same number of calls. Though, it is difficult to have a rigorous and clean argument, especially for ciphers with 3 calls. The most intriguing direction is thus to address fully general 2- and 3-call blockciphers, which may shed more lights on iterations. Another intriguing question is whether there are smart ideas to unify the complicated cases in $E3$ analysis. Influences of other aspects such as memory restrictions on adversaries and simulators are also of interest.

On the constructive side, it is intriguing to study the achievability of computational indifferentiability with 3 calls: hardness assumptions on graph problems or key derivation functions might be helpful.

Unlike most practice in symmetric cryptography, our Theorem 3 is asymptotic. The key issue is that our differentiator has to "know" the simulator limitations for its case decision. Classically, simulator (query) complexity is only *polynomially bounded* and seems incompatible with concrete treatments. Fully concrete characterizations may need a new paradigm and are left for future work.

Roadmap. We serve notations and definitions in Sect. 2. Then, as mentioned, we provide a technical overview in Sect. 3.

For the main elaborations, we first formalize the *Fundamental Properties* in Sect. 4. We then give detailed elaborations and characterizations for our 1-call cipher $E1$ as well as its main result in Sect. 5. Our main results on $E2$ and $E3$

are then given in Sect. 6 and 7 respectively. Due to space constraints, detailed proofs are mostly deferred to the full version [28].

2 Preliminaries

Fix n as the security parameter, and write $\mathrm{poly}(n)$ and $\mathrm{negl}(n)$ for arbitrary polynomial and negligible functions respectively. Denote by $\perp$ the empty string. Given $x \in \{0,1\}^n$ and $a \leq n$, denote by $\mathsf{left}_a(x)$ (resp., $\mathsf{right}_a(x)$) the a leftmost (resp., rightmost) bits of x. When two sets A and B are disjoint, we denote $A \sqcup B$ their (disjoint) union. For any domain, denote by id the identity function.

An m-bit random permutation is a permutation that is uniformly selected from $\mathsf{Perm}(m)$, the set of all $(2^m)!$ possible m-bit permutations. Throughout the remaining, we denote by $\mathbf{IC} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ an ideal cipher (which is randomly picked from all (κ, n)-blockciphers) with $\kappa = \mathrm{poly}(n)$.

Permutation family $\mathcal{P}$. As briefed in the Introduction, we consider constructing an n-bit blockcipher from a permutation family oracle $\mathcal{P}$ that "provides all but the goal". In detail, $\mathcal{P} = (\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_{|\mathcal{I}|})$ provides independent random permutations indexed by $i \in \mathcal{I}$, where $\mathbf{P}_i \in \mathsf{Perm}(m(i))$ for a fixed function $m : \mathcal{I} \to \mathrm{poly}(n)$ (viewed as parameters of $\mathcal{P}$). It can be $m(i) \gg n$ for some $i \in \mathcal{I}$. The index set is thus partitioned as $\mathcal{I} = \mathcal{I}_{\leq n} \sqcup \mathcal{I}_{>n}$, where $i \in \mathcal{I}_{\leq n}$ if and only if $m(i) \leq n$. We require $|\mathcal{I}_{\leq n}| = O(\mathrm{poly}(n))$, while $\mathcal{I}_{>n}$ can be exponentially large. We call permutations with width$> n$ *wide*. Denote by $m_{max} := \max_{i \in \mathcal{I}} m(i)$ and $m_{min} := \min_{i \in \mathcal{I}} m(i)$ the size of largest, resp. smallest permutation in $\mathcal{P}$.

Oracle $\mathcal{P}$ accepts queries of the form (i, δ, z), where $i \in \mathcal{I}$ is the index, $\delta \in \{+, -\}$ indicates if forward $\mathbf{P}_i$ or backward $\mathbf{P}_i^{-1}$ is queried, and $z \in \{0,1\}^{m(i)}$ is the actual $m(i)$-bit input. For $\delta \in \{+, -\}$, we denote $\overline{\delta}$ the opposite of δ.

Indifferentiability. Let $E^{\mathcal{P}}$ be a cryptographic construction that internally queries $\mathcal{P}$, $\mathbf{IC}$ be the ideal crypto object of $E^{\mathcal{P}}$, and $S^{\mathbf{IC}}$ be a simulator that queries $\mathbf{IC}$ and provides the same interfaces as $\mathcal{P}$. Then, for any distinguisher D, the indifferentiability advantage of D against $E^{\mathcal{P}}$ is

$$\mathrm{Adv}^{indif}_{E^{\mathcal{P}}, \mathbf{IC}, S}(D) = \left| \Pr\left[D^{E^{\mathcal{P}}, \mathcal{P}} = 1\right] - \Pr\left[D^{\mathbf{IC}, S^{\mathbf{IC}}} = 1\right] \right|.$$

$E^{\mathcal{P}}$ is indifferentiable (in the asymptotic sense), as long as for any polynomial-query D: (a) the advantage $\mathrm{Adv}^{indif}_{E^{\mathcal{P}}, \mathbf{IC}, S}(D)$ is $\mathrm{negl}(n)$ w.r.t. the security parameter n for any D, and (b) the number of queries made by S to $\mathbf{IC}$ is $\mathrm{poly}(n)$.

Notations for differentiators. Consider blockciphers $E1^{\mathcal{P}}, E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$ built upon the oracle $\mathcal{P}$. By the above, to break indifferentiability, we shall exhibit a differentiator D that "fools" any query-efficient simulator $S^{\mathbf{IC}}$ with non-negligible probability. Notice, D has access to two oracles (E, P) where $\mathrm{E} \in \{E1^{\mathcal{P}}, E2^{\mathcal{P}}, E3^{\mathcal{P}}, \mathbf{IC}\}$ and $\mathrm{P} \in \{\mathcal{P}, S^{\mathbf{IC}}\}$. To describe the interaction between $D^{\mathrm{E},\mathrm{P}}$ and its oracles E, P, we use the expressions $\mathrm{P}(i, \delta, z) \to z'$ to mean that D queries P on (i, δ, z) and P answers with z', and $\mathrm{E}(K, x) \to y$ to mean that E is queried on (K, x) and returns y. Note that in the latter case, the query may be made by S. The notation $\mathrm{E}^{-1}(K, y) \to x$ is similar.

As convention, our differentiators always output 1 when it guesses the "real world", and output 0 when it guesses the "ideal" or "simulated world".

Tools from Extremal Graph Theory. Consider a bipartite graph $\mathcal{G} = (\mathcal{V}_L, \mathcal{V}_R, \mathcal{E})$. Intuitively, if $|\mathcal{E}|$ is sufficiently large, then $\mathcal{G}$ must have short cycles (since long cycles will be truncated). This was proven by Hoory [32], and will help establishing the existence of certain structures. To ease applying, we restate Hoory's result [32, Eqs. (1) and (2)] as follows.

Proposition 1. *Let $\mathcal{G} = (\mathcal{V}_L, \mathcal{V}_R, \mathcal{E})$ be a bipartite graph such that:*

(i) $|\mathcal{V}_L|$ and $|\mathcal{V}_R|$ have a common upper bound, i.e., there exists an integer $M > 0$ such that $|\mathcal{V}_L| \leq M$, $|\mathcal{V}_R| \leq M$; and
(ii) $|\mathcal{E}| \geq \big((M)^{\frac{1}{t-1}} + 1\big) \times M$ for some positive integer t.

Then, $\mathcal{G}$ contains a cycle $C_{2\ell}$ with $\ell \leq t$.

If $|\mathcal{E}|$ is large, then $\mathcal{G}$ contains a small complete bipartite graph (a.k.a. biclique). This was proven by Kővári, Sós and Turán [38], and is restated as follows.

Proposition 2. *Let $\mathcal{G} = (\mathcal{V}_L, \mathcal{V}_R, \mathcal{E})$ be a bipartite graph such that:*

(i) There exist two integers $M, N > 0$ such that $|\mathcal{V}_L| \leq M$ and $|\mathcal{V}_R| \leq N$; and
(ii) $|\mathcal{E}| \geq (b-1)^{\frac{1}{a}} \cdot MN^{1-\frac{1}{a}} + (a-1)N$.

Then, $\mathcal{G}$ contains the complete bipartite graph $K_{a,b}$ as a sub-graph.

We refer to the full version [28] for how to concretely derive the two propositions.

3 Technical Overview

As mentioned, we characterize the 1-call model $E1^{\mathcal{P}}$ and then extend the discussion to 2- and 3-call iterated models $E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$. Below in Sect. 3.1, we first elaborate more on *Fundamental Properties* underlying our reasoning. We then provide intuitions for $E1^{\mathcal{P}}$, $E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$ in Sect. 3.2, 3.3 and 3.4 in turn.

3.1 Fundamental Properties

As mentioned, our analyses rely on four properties that we believe fundamental to blockcipher oracle procedures. First, the *definition of the notion of blockciphers* yield two properties for a blockcipher oracle procedure $E^{\mathcal{P}}$:

(i) **Efficient invertibility**: blockciphers should be efficiently invertible. Namely, there is a corresponding oracle procedure $(E^{-1})^{\mathcal{P}}$ computing its inverse;
(ii) **Deterministic**: blockciphers should be *deterministic*. For $E^{\mathcal{P}}$, it means for a fixed oracle $\mathcal{P}$, *evaluating $E^{\mathcal{P}}(K, x) \to y$ and the corresponding decipherment $(E^{-1})^{\mathcal{P}}(K, y)$ always yield the same transcript of $\mathcal{P}$-queries and responses.*

Besides, since an oracle procedure $E^{\mathcal{P}}$ shall have a fixed description that is independent from $\mathcal{P}$, sub-procedures in $E^{\mathcal{P}}$ are **oracle-independent**. We further assume that $E^{\mathcal{P}}$ is **non-degenerate** and cannot be "simplified" in terms of $\mathcal{P}$ calls, i.e., no encipherment $E^{\mathcal{P}}(K, x)$ can be approximately computed using less $\mathcal{P}$ calls than $E^{\mathcal{P}}$. Formal definitions will be given in Sect. 4.

3.2 Full characterization of 1-call cipher $E1$

As per mentioned, $E1^{\mathcal{P}}(K, x) := \varphi^{out}\big(K, \mathcal{P}(\varphi^{in}(K, x))\big)$ and $(E1^{-1})^{\mathcal{P}}(K, y) := \gamma^{out}\big(K, \mathcal{P}(\gamma^{in}(K, y))\big)$, where φ^{in}, φ^{out}, γ^{in} and γ^{out} can be arbitrary oracle-independent functions. The *Fundamental Properties* already ensure a number of non-trivial properties (on oracle procedures of blockciphers).

Inv-freeness and its oracle-independence. Our first observation is about inverse-freeness of $E1^{\mathcal{P}}$. An encipherment $E1^{\mathcal{P}}(K, x)$ is *inverse-free* (inv-free for short), if $E1^{\mathcal{P}}(K, x) \to y$ and its corresponding decipherment $(E1^{-1})^{\mathcal{P}}(K, y) \to x$ call $\mathcal{P}(i, \delta, \star)$ on the same direction δ; otherwise, $E1^{\mathcal{P}}(K, x)$ is *non-inverse-free* (non-inv-free). In common designs (Feistel, Misty, IEM, etc.), encipherments under a fixed key are either all inv-free or non-inv-free for all plaintext. However, in general, the inv-freeness of $E1^{\mathcal{P}}(K, x)$ may depend on x, admitting *data-dependent inv-freeness*. We serve an example in Fig. 3.

Our observation is that *in $E1^{\mathcal{P}}$, inv-freeness cannot depend on the oracle $\mathcal{P}$*, i.e., one can decide if an encipherment $E1^{\mathcal{P}}(K, x)$ is inv-free without querying $\mathcal{P}$. Intuitively, it is because the query directions of encipherment and decipherment are determined by the input functions φ^{in} and γ^{in}, which are oracle-independent. The formal presentation will be given in Lemma 1. As will be seen, exploitable weakness in an encipherment $E1^{\mathcal{P}}(K, x)$ depends on its inv-freeness, the oracle-independence of which turns out crucial in our attacks.

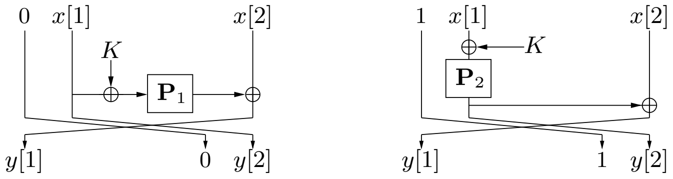

Fig. 3. A 1-call cipher/round that has data-dependent inverse-freeness. (Left) when the leftmost bit of $x = 0\|x[1]\|x[2]$ is 0 (a permutation-based Feistel round); (Right) when the leftmost bit of $x = 1\|x[1]\|x[2]$ is 1 (a Misty-like round).

Properties of inv-free $\boldsymbol{E1^{\mathcal{P}}(K, x)}$. For intuitions, consider the key-prepended Feistel round $y = \Psi^{\mathbf{F}}(K, x) := \big(\mathsf{right}_{n/2}(x) \oplus \mathbf{F}(K\|\mathsf{left}_{n/2}(x))\big)\|\mathsf{left}_{n/2}(x)$. This round is inv-free: an encipherment $\Psi^{\mathbf{F}}(K, x)$ and its corresponding decipherment $(\Psi^{-1})^{\mathbf{F}}(K, y)$ make the same "forward" call to $\mathbf{F}(z)$, $z = K\|\mathsf{left}_{n/2}(x) = K\|\mathsf{right}_{n/2}(y)$. This means some information of x is kept in the ciphertext y

without "protection". The same property is shared by various Feistel variants [2, Chapter 1.3.1] (including the Lai-Massey scheme [2, Chapter 1.5]). Casting it into our general model $E1^{\mathcal{P}}$, it means *inv-free* $E1^{\mathcal{P}}(K,x) \rightarrow y$ *must have* $\varphi^{in}(K,x) = \gamma^{in}(K,y)$.

As a less obvious fact in $\Psi^{\mathbf{F}}$, there necessarily exist many distinct encipherments that make the same $\mathbf{F}$-call. I.e., $\Psi^{\mathbf{F}}(K,x)$ calls $\mathbf{F}(z)$ as long as $\mathsf{right}_{n/2}(x) = \mathsf{right}_{n/2}(z)$, and there are $2^{n/2}$ possible x for every z. Similarly for other inv-free designs. It turns out that: *with the non-degeneracy assumption on* $E1$, *if there is one inv-free* $E1^{\mathcal{P}}(K,x)$ *then there are* $\Omega(poly(n))$ *distinct inv-free* $E1^{\mathcal{P}}(K,x_1)$, $E1^{\mathcal{P}}(K,x_2)$, ... *that collide on the* $\mathcal{P}$*-call, i.e.,* $\varphi^{in}(K,x_1) = \varphi^{in}(K,x_2) = ... = \varphi^{in}(K,x)$. This further implies that under each key K, all inv-free $E1^{\mathcal{P}}(K,x)$ give rise to $o\big(\frac{2^n}{\mathrm{poly}(n)}\big)$ distinct $\mathcal{P}$-calls (even if they can query exponentially many permutations with width$> n$).

We refer to Lemmas 2–4 in Sect. 5.2 for formal elaborations.

Properties of non-inv-free $\boldsymbol{E1^{\mathcal{P}}(K,x)}$. For intuitions, consider the IEM round $y = \mathrm{EM}^{\mathbf{P}}(K,x) := K \oplus \mathbf{P}(K \oplus x)$, which is non-inv-free since $(\mathrm{EM}^{-1})^{\mathbf{P}}$ always calls $\mathbf{P}^{-1}$. $\mathrm{EM}^{\mathbf{P}}$ is more secure than $\Psi^{\mathbf{F}}$. In fact, attacks against $\mathrm{EM}^{\mathbf{P}}$ have to exploit at least 2 keys K, K' [1, Sect. 3.1, full version] and seek for encipherments $\mathrm{EM}^{\mathbf{P}}(K,x)$ and $\mathrm{EM}^{\mathbf{P}}(K',x')$ that collide on the $\mathbf{P}$-call, i.e., with $K \oplus x = K' \oplus x'$. Such collided encipherments *do exist*, because *EM*$^{\mathbf{P}}$ *cannot use wide* $\mathbf{P}$. Concretely, to invoke a wide $\mathbf{P}$, $\mathrm{EM}^{\mathbf{P}}$ must pad $x \in \{0,1\}^n$ with some "non-trivial" information (e.g., $\mathbf{P}(x\|0)$, or $\mathbf{P}(x\|K)$); but then, by invoking $\mathbf{P}^{-1}$, decipherments are unlikely to "recover" correctly padded $\mathbf{P}$-inputs. In fact, this irrecoverability is the core idea of *Encode-then-Encipher* [5].

It turns out that this irrecoverability stems from oracle-independence. In detail, assuming oracle-independence of φ^{in} and γ^{in}, *non-inv-free encipherments* $E1^{\mathcal{P}}(K,x)$ *can only query permutations with width*$\leq n$. Thus, non-inv-free give rise to at most $|\mathcal{I}_{\leq n}|2^{n+1}$ distinct $\mathcal{P}$-calls. We refer to Lemma 5 in Sect. 5.2 for formal elaborations.

Attack $\boldsymbol{E1^{\mathcal{P}}}$. With the above properties, we are able to bump into our differentiator $D1$ on $E1^{\mathcal{P}}$. In detail, the cipher $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \rightarrow \{0,1\}^n$ may fall into two cases.

Case 1: there exists at least 1 inv-free encipherment $E1^{\mathcal{P}}(K,x)$. As discussed, this means we can find $t = \Omega(\mathrm{poly}(n))$ distinct inv-free $E1^{\mathcal{P}}(K,x_1), ..., E1^{\mathcal{P}}(K,x_t)$ that make the same $\mathcal{P}$-call $\mathcal{P}(i,\delta,z)$, $(i,\delta,z) = \varphi^{in}(K,x_1) = ... = \varphi^{in}(K,x_t)$. Thus, the restriction of $E1^{\mathcal{P}}(K,\cdot)$ to $\{x_1,...,x_t\}$ is a bijection defined upon a polynomial-length random string $z' = \mathcal{P}(i,\delta,z)$, and we can apply an entropy-based differentiating approach [41].

Case 2: $E1^{\mathcal{P}}(K,x)$ *is non-inv-free for all* $(K,x) \in \mathcal{K} \times \{0,1\}^n$. Then, $E1^{\mathcal{P}}(K,x)$ can only invoke the permutations in $\mathcal{P}$ with width$\leq n$ (as discussed). Therefore, the number of possible images of φ^{in} is at most $|\mathcal{I}_{\leq n}|2^{n+1}$. As long as $|\mathcal{K}|2^n \geq |\mathcal{I}_{\leq n}|2^{n+1}$, i.e., the keyspace has $|\mathcal{K}| \geq 2|\mathcal{I}_{\leq n}|+1$ (which is $O(\mathrm{poly}(n))$, though), the pigeonhole principle guarantees the existence of $(K,x), (K',x') \in \mathcal{K} \times \{0,1\}^n$

with collision $\varphi^{in}(K,x) = \varphi^{in}(K',x')$. $D1$ thus finds such a pair of collided $(K,x),(K',x')$ and attacks by checking if $\gamma^{in}(K,\mathrm{E}(K,x)) = \gamma^{in}(K',\mathrm{E}(K',x'))$.

The formal proof, deferred to the full version [28], is more technical and relies on a sort of "regularity" of the input functions φ^{in} and γ^{in} (Lemma 6).

3.3 Attack 2-call iterated cipher $E2$

Built upon our above results on $E1^{\mathcal{P}}$, we further consider our 2-call model $E2^{\mathcal{P}}$. Recall that the keyspace $\mathcal{K}$ of $E2^{\mathcal{P}}$ can be partitioned $\mathcal{K} = \mathcal{K}^{(0)} \sqcup \mathcal{K}^{(1)}$, such that $E2^{\mathcal{P}}(K,x) = \Pi_3^{\mathcal{P}}(K\|\mathsf{kd}^{\mathcal{P}}(K),x)$ for all $K \in \mathcal{K}^{(1)}$, whereas $E2^{\mathcal{P}}(K,x) = \Pi_2^{\mathcal{P}}\big(K,\Pi_1^{\mathcal{P}}(K,x)\big)$ for all $K \in \mathcal{K}^{(0)}$. The sub-procedures $\mathsf{kd}^{\mathcal{P}}$ has $\mathsf{kd}^{\mathcal{P}}(K) = \mathcal{P}(f(K))$ for another oracle-independent function f. In addition, for $j = 1,2,3$, $\Pi_j^{\mathcal{P}}$ is a 1-call cipher with input and output functions $\varphi_j^{in}, \varphi_j^{out}, \gamma_j^{in}$ and γ_j^{out}. We refer to Fig. 2 for illustration and Fig. 11 for a pseudocode description.

This model $E2^{\mathcal{P}}$ may fall into two cases.

Case 1: $E2^{\mathcal{P}}$ invokes kd for sufficiently many keys. Formally, if the key sets have $|\mathcal{K}^{(1)}| \geq 2|\mathcal{I}_{\leq n}| + 1$, we simply pick $\lambda = 2|\mathcal{I}_{\leq n}| + 1$ keys $K_1, ..., K_\lambda \in \mathcal{K}^{(1)}$ and derive subkeys $s_1 = \mathsf{kd}^{\mathcal{P}}(K_1), ..., s_\lambda = \mathsf{kd}^{\mathcal{P}}(K_\lambda)$. This consumes at most $\lambda = O(\mathrm{poly}(n))$ P-queries. We then view the round $\Pi_3^{\mathcal{P}}$ as a 1-call cipher with keyspace $\{K_1\|s_1, ..., K_\lambda\|s_\lambda\}$ and apply our differentiator $D1$. (It is thus crucial that $D1$ can break $E1$ with polynomial-keyspace.)

Case 2: $E2^{\mathcal{P}}$ is 2-iteration for sufficiently many keys. The concrete condition is $|\mathcal{K}^{(0)}| \geq \big(6\big(3|\mathcal{I}_{\leq n}|\big)^{\frac{1}{n}} + 3\big)|\mathcal{I}_{\leq n}| = O(\mathrm{poly}(n))$. Our idea (non-trivially) generalizes existing specific attacks, which is elaborated as follows.

Outset: boomerang property. Our initial intuition lies in a chosen-key boomerang differentiator against the 2-round IEM cipher $y = K \oplus \mathbf{P}_2(K \oplus \mathbf{P}_1(K \oplus x))$, $K \in \{0,1\}^n$ (which is motivated by Andreeva et al.'s [1, Sect. 3.2, full version]). Briefly, for any x, let $u = \mathbf{P}_1(K \oplus x)$. The attack begins by computing four distinct pairs $(K_1,u_1),(K_2,u_2),(K_3,u_3),(K_4,u_4)$ with $u_1 = u_2$, $u_3 = u_4$; $K_1 \oplus u_1 = K_3 \oplus u_3$ and $K_2 \oplus u_2 = K_4 \oplus u_4$. I.e., they induce two collided inputs to $\mathbf{P}_1^{-1}$ and two collide inputs to $\mathbf{P}_2$. Once such four pairs are derived, the differentiator can computes a 4-tuple of cipher inputs/outputs $\big((K_1,x_1,y_1), ..., (K_4,x_4,y_4)\big)$ that has $K_1 \oplus x_1 = K_2 \oplus x_2$, $K_3 \oplus x_3 = K_4 \oplus x_4$; $K_1 \oplus y_1 = K_3 \oplus y_3$, $K_2 \oplus y_2 = K_4 \oplus y_4$; as shown in Fig. 4 (left). Such a 4-tuple satisfies an evasive relation [39] and is hard to found in the ideal world. Actually the involved structure is the basis of the boomerang attack developed in [49].

A similar boomerang can be exhibited in the 2-round Feistel. Motivated by these, our differentiator against the general 2-iteration cipher tries to find pairs $(K_1,u_1),(K_2,u_2),(K_3,u_3),(K_4,u_4) \in \mathcal{K}^{(0)} \times \{0,1\}^n$ that induce similar collided $\mathcal{P}$-calls, i.e., $\gamma_1^{in}(K_1,u_1) = \gamma_1^{in}(K_2,u_2)$, $\gamma_1^{in}(K_3,u_3) = \gamma_1^{in}(K_4,u_4)$; $\varphi_2^{in}(K_1,u_1) = \varphi_2^{in}(K_3,u_3)$ and $\varphi_2^{in}(K_2,u_2) = \varphi_2^{in}(K_4,u_4)$, as shown in Fig. 4 (right). This is a general boomerang property. Unlike Fig. 4 (left), the four encipherments may not be non-inv-free in two rounds: actually, Fig. 4 (right) serves an example where the 1st round of $E2^{\mathcal{P}}(K_3,x_3)$ and $E2^{\mathcal{P}}(K_4,x_4)$ are inv-free.

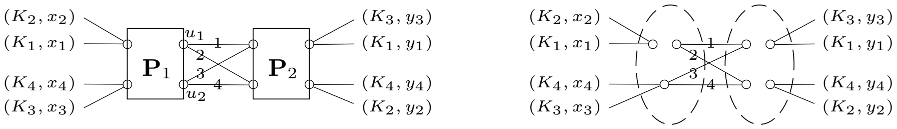

Fig. 4. (Top) Boomerang distinguisher in 2-round IEM. Circles indicate values in domain and range of $\mathbf{P}_1$ and $\mathbf{P}_2$ (in particular, u_1 and u_2 are marked), and lines indicate encipherment flows. To simplify, for lines between $\mathbf{P}_1$ and $\mathbf{P}_2$ the pair (K_j, u_j) is simplified as j. (Bottom) An example of boomerang distinguisher in 2-round general $E2$. Circles indicate values in domain and range of $\mathcal{P}$, and lines indicate encipherment flows. When a line "crosses" a pair of circles, it means the encipherment is non-inv-free in that round, and the two circles (naturally) indicate the $\mathcal{P}$ inputs and outputs; when a line "crosses" a single circle, it means the encipherment is non-inv-free in that round (so that rightward and leftward evaluations reach the same $\mathcal{P}$ input). Thus, the four encipherments are all non-inv-free in the 2nd round; in the 1st round, $E2^{\mathcal{P}}(K_1, x_1)$ and $E2^{\mathcal{P}}(K_2, x_2)$ are non-inv-free, while $E2^{\mathcal{P}}(K_3, x_3)$ and $E2^{\mathcal{P}}(K_4, x_4)$ are inv-free.

From boomerang to yoyo. But does such a 4-tuple ever exist? Unlike "concrete" ciphers such as IEM and Feistel, this is unclear in $E2$. To solve this, we apply Hoory's [32] result on girth (i.e., maximal length of cycles in a graph). Briefly, if we view the possible inputs to $\mathcal{P}$ as shores and the pairs $(K, u) \in \mathcal{K}^{(0)} \times \{0,1\}^n$ as edges, then we can build a bipartite graph $\mathcal{G}$, and the above 4-tuple becomes a 4-cycle C_4 (i.e., cycle of length 4) in $\mathcal{G}$. By Hoory [32] (which is restated in Sect. 2, Proposition 1), as long as the number of edges is large enough, such a 4-cycle or 4-tuple is guaranteed to exist.

However, as will be clear in the analysis (available in [28]), the above requires $\mathcal{K}^{(0)}$ to be of exponential size, which would prohibit its application in our later attack against $E3$. To remedy, we consider longer cycles $C_{2\lambda}$, $\lambda \leq n+1$. I.e., our differentiator seeks for a 2λ-tuple $\big((K_1, u_1), ..., (K_{2\lambda}, u_{2\lambda})\big)$ that has

$$\begin{aligned}
&\varphi_2^{in}(K_1, u_1) = \varphi_2^{in}(K_2, u_2), && \gamma_1^{in}(K_2, u_2) = \gamma_1^{in}(K_3, u_3),\\
&\varphi_2^{in}(K_3, u_3) = \varphi_2^{in}(K_4, u_4), && \gamma_1^{in}(K_4, u_4) = \gamma_1^{in}(K_5, u_5), \quad ...\\
&\varphi_2^{in}(K_{2\lambda-1}, u_{2\lambda-1}) = \varphi_2^{in}(K_{2\lambda}, u_{2\lambda}), && \gamma_1^{in}(K_{2\lambda}, u_{2\lambda}) = \gamma_1^{in}(K_1, u_1). \qquad (2)
\end{aligned}$$

Once such a 2λ-tuple is found, our differentiator can computes a 2λ-tuple of $E2$ inputs/outputs $\big((K_1, x_1, y_1), ..., (K_{2\lambda}, x_{2\lambda}, y_{2\lambda})\big)$ that has a "cycle of collisions". I.e., $\gamma_2^{in}(K_1, y_1) = \gamma_2^{in}(K_2, y_2), \varphi_1^{in}(K_2, x_2) = \varphi_1^{in}(K_3, x_3), ..., \gamma_2^{in}(K_{2\lambda-1}, y_{2\lambda-1}) = \gamma_2^{in}(K_{2\lambda}, y_{2\lambda}), \varphi_1^{in}(K_{2\lambda}, x_{2\lambda}) = \varphi_1^{in}(K_1, x_1)$. An example with $\lambda = 4$ is shown in Fig. 5. This is actually a general version of the yoyo distinguisher [45]. By Hoory [32], $|\mathcal{K}^{(0)}| \geq \big(6\big(3|\mathcal{I}_{\leq n}|\big)^{\frac{1}{n}} + 3\big)|\mathcal{I}_{\leq n}| = O(\mathrm{poly}(n))$ already suffices for the existence of $\big((K_1, u_1), ..., (K_{2\lambda}, u_{2\lambda})\big)$. Note that Hoory does not apply when $\mathcal{G}$ is a multigraph, but this implies existence of C_2. These solve our first problem.

Non-degenerate input functions. Subtleties remain. To argue that no polynomial-query simulator can work out a similar 2λ-tuple of ideal cipher inputs/outputs $\big(\mathbf{IC}(K_1, x_1) = y_1, ..., \mathbf{IC}(K_{2\lambda}, x_{2\lambda}) = y_{2\lambda}\big)$, the input functions φ_1^{in} and γ_2^{in}

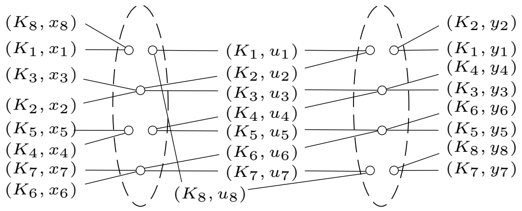

Fig. 5. An example of general yoyo distinguisher with $\lambda = 4$ in $E2$. Meanings of the objects follow Fig. 4.

must be somewhat "non-degenerate". Roughly, $\Pr[x \xleftarrow{\$} \{0,1\}^n : \varphi_1^{in}(K,x) = (i,\delta,z)] = \mathrm{negl}(n)$ and $\Pr[y \xleftarrow{\$} \{0,1\}^n : \gamma_2^{in}(K,y) = (i,\delta,z)] = \mathrm{negl}(n)$ for any K and any (i,δ,z).

Wlog consider φ_1. Indeed, due to the aforementioned "regularity" (Lemma 6), it can be proven $\Pr_x[\varphi_1^{in}(K,x) = (i,\delta,z) \mid \Pi_1^{\mathcal{P}}(K,x)$ non-inv-free$] = \mathrm{negl}(n)$. But $\varphi_1^{in}(K,\cdot)$ may lead $\Omega(2^n/\mathrm{poly}(n))$ distinct inv-free $\Pi_1^{\mathcal{P}}(K,x)$ to the same call $\mathcal{P}(i,\delta,z)$ (i.e., being highly biased), which enables the simulator to cheat.

A complete case-study thus has to consider whether φ_1^{in} and γ_2^{in} are "non-degenerate". However, input functions in virtually all blockciphers are indeed "non-degenerate": otherwise, the round is ridiculously weak. Meanwhile, complete case-study would take us quite far afield. We thereby decide to simplify and introduce *non-degenerate input functions* as an additional assumption for $E2^{\mathcal{P}}$ and $E3^{\mathcal{P}}$, i.e., $\Pr_x[\varphi_1^{in}(K,x) = (i,\delta,z) \mid \Pi_1^{\mathcal{P}}(K,x)$ inv-free$] = \mathrm{negl}(n)$ and $\Pr_y[\gamma_2^{in}(K,y) = (i,\delta,z) \mid (\Pi_2^{-1})^{\mathcal{P}}(K,y)$ inv-free$] = \mathrm{negl}(n)$. We refer to Sect. 6 for more details. With this additional restriction, we prove that no polynomial-query simulator can work out the aforementioned 2λ-tuple. In fact, Eq. (2) defines a novel evasive relation in 2-round general ciphers, which is stronger than differentiability. See Sect. 6 for formal result and [28] for detailed analysis.

It is crucial to restrict our discussion to iterated blockciphers: since the set of valid intermediate values u between the rounds is simply $\{0,1\}^n$, an attacker can pick such a u and compute forward or backward. Indeed, this middle-to-sides approach is common in known- and chosen-key attacks [37].

3.4 Attack 3-call iterated cipher $E3$

We further consider our 3-call model $E3^{\mathcal{P}}$. Recall that $E3^{\mathcal{P}} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ has $\kappa = \Theta(\mathrm{poly}(n))$, and its keyspace can be partitioned $\{0,1\}^\kappa = \mathcal{K}^{(0)} \sqcup \mathcal{K}^{(1)} \sqcup \mathcal{K}^{(2)}$, such that:

(i) $E3^{\mathcal{P}}(K,x) = \Pi_6^{\mathcal{P}}(K \| \mathsf{kd}_1^{\mathcal{P}}(K), x)$ for all $K \in \mathcal{K}^{(2)}$;
(ii) $E3^{\mathcal{P}}(K,x) = \Pi_5^{\mathcal{P}}\big(K \| \mathsf{kd}_2^{\mathcal{P}}(K), \Pi_4^{\mathcal{P}}(K \| \mathsf{kd}_2^{\mathcal{P}}(K), x)\big)$ for all $K \in \mathcal{K}^{(1)}$;
(iii) $E3^{\mathcal{P}}(K,x) = \Pi_3^{\mathcal{P}}\big(K, \Pi_2^{\mathcal{P}}\big(K, \Pi_1^{\mathcal{P}}(K,x)\big)\big)$ for all $K \in \mathcal{K}^{(0)}$.

The sub-procedures $\mathsf{kd}_1^{\mathcal{P}}(K)$ and $\mathsf{kd}_2^{\mathcal{P}}(K)$ derive corresponding subkeys via two and one calls to $\mathcal{P}$ respectively. In addition, for $j = 1, 2, ..., 6$, $\Pi_j^{\mathcal{P}}$ is a 1-call cipher with input and output functions $\varphi_j^{in}, \varphi_j^{out}, \gamma_j^{in}$ and γ_j^{out}. We refer to Fig. 12 for pseudocode of $E3^{\mathcal{P}}$.

When $E3^{\mathcal{P}}$ invokes kd_1 or kd_2 for sufficiently many keys $K \in \{0,1\}^\kappa$, we again derive poly(n) subkeys to reduce $E3^{\mathcal{P}}$ to $E1$ or $E2$ instances with polynomial keyspace, and apply our previous differentiators (thanks to that our differentiators break $E1$ and $E2$ with polynomial keyspace).

The crux is the case where $E3^{\mathcal{P}}(K,x) = \Pi_3^{\mathcal{P}}\big(K, \Pi_2^{\mathcal{P}}\big(K, \Pi_1^{\mathcal{P}}(K,x)\big)\big)$ for virtually all 2^κ keys K. Depending on whether the $\Theta(2^{\kappa+n})$ encipherments are "mostly" inv-free or not in the 3 rounds, exploitable non-random properties significantly vary in the $2^3 = 8$ cases and cannot be unified. We thereby have to appeal for a (lengthy) case-study.

Furthermore, note that inv-freeness can be data-dependent, which causes a subtle technical challenge. Namely, without querying P, one cannot fully decide if a certain encipherment is inv-free in the 3 rounds.[7] But querying P would trigger simulator actions in the ideal world, and the simulated P may be defined to change the inv-freeness of the encipherments in question. This turns out a technical challenge, and we call it *decisional inv-free problem*. Our solution is two-fold. First, we identified relevant conditions that are decidable without querying P, so that our differentiator could invoke the right subroutine for case-study without attracting simulator's attention. Meanwhile, to compute (intermediate) values of the encipherments in question, our differentiator (tries the best to) query the enciphering oracle E instead of P to avoid "waking" the simulator. Our case conditions ensure that the ideal cipher responses (in the ideal world) will satisfy our expectations on inv-freeness. We will elaborate more later.

Below we denote by $x \in \{0,1\}^n$ the plaintext, $u = \Pi_1^{\mathcal{P}}(K,x)$ the 1st round output, $w = \Pi_2^{\mathcal{P}}(K,u)$ the 2nd round output and $y = \Pi_3^{\mathcal{P}}(K,w)$ the ciphertext.

Case 1: there are $\boldsymbol{\Theta(2^\kappa)}$ keys $\boldsymbol{K}$ s.t. only $\boldsymbol{o(2^n/\mathrm{poly}(n))}$ $\boldsymbol{\Pi_1^{\mathcal{P}}(K,x)}$ are non-inv-free, and only $\boldsymbol{o(2^n/\mathrm{poly}(n))}$ $\boldsymbol{(\Pi_3^{-1})^{\mathcal{P}}(K,y)}$ are non-inv-free. Roughly, this means most of the $\Theta(2^{\kappa+n})$ encipherments $E3^{\mathcal{P}}(K,x)$ are inv-free in 1st and 3rd rounds. A famous example is the 3-round Feistel $\mathrm{Feistel3}(K,x) := \Psi^{\mathbf{F}_3}\big(K, \Psi^{\mathbf{F}_2}\big(K, \Psi^{\mathbf{F}_1}(K,x)\big)\big)$. Since the 2nd round could be arbitrary, the "hybrid" cipher $\mathrm{Hyb}(K,x) := \Psi^{\mathbf{F}_3}\big(K, K \oplus \mathbf{P}\big(K \oplus \Psi^{\mathbf{F}_1}(K,x)\big)\big)$ is another example. A fact shared by the two examples is that *there are many encipherments that collide on* $\mathbf{F}$*- or* $\mathbf{P}$*-calls in the 2nd round.* Concretely,

- In Feistel3, let $u = \Psi^{\mathbf{F}_1}(K,x)$. Then, for any $z \in \{0,1\}^{n/2}$, all the $2^{n/2}$ encipherments with key K and 1st round output $u = \star \| z$ call $\mathbf{F}_2(K\|z)$;
- In Hyb1, let $u = \Psi^{\mathbf{F}_1}(K,x)$. Then, for any $z \in \{0,1\}^n$, all the 2^n encipherments with (K,u), $K \oplus u = z$, call $\mathbf{P}(z)$.

It turns out that this can be proven in the general 3-round cipher (in this case): there exist $t = \Omega(\mathrm{poly}(n))$ distinct intermediate values $(K_1,u_1), ..., (K_t,u_t)$

[7] E.g., given K and a 1st round output $u \in \{0,1\}^n$, one can decide the inv-freeness of the corresponding encipherment in the 1st and 2nd rounds, since $\Pi_1^{\mathcal{P}}(K,u)$ and $(\Pi_2^{-1})^{\mathcal{P}}(K,u)$ can be decided. But without querying P, one cannot derive $w = \Pi_2^{\mathcal{P}}(K,u)$, and thus cannot decide if the process is inv-free in the 3rd round.

that collide on 2nd round $\mathcal{P}$-call, i.e., $\varphi_2^{in}(K_1,u_1)=...=\varphi_2^{in}(K_t,u_t)=(i_2,\delta_2,z_2)$, as shown in Fig. 6 (left).

With such a "star" structure, we issue the "central" query $\mathrm{P}(i_2,\delta_2,z_2)\to z_2'$. In the real world, the response z_2' is consistent with $\Omega(\mathrm{poly}(n))$ encipherments. Namely, for all $j\in\{1,...,t\}$, suppose we evaluate $w_j\leftarrow\varphi_2^{out}(K_j,z_2',u_j)$, $x_j\leftarrow(\Pi_1^{-1})^{\mathrm{P}}(K_j,u_j)$ and $\mathrm{E}(K_j,x_j)\to y_j$. In the real world, if $\Pi_3^{\mathrm{P}}(K_j,w_j)$ is inv-free then it holds $\varphi_3^{in}(K_j,w_j)=\gamma_3^{in}(K_j,y_j)$. In the ideal world, S (roughly) has to find ideal cipher inputs/outputs $\mathbf{IC}(K_j,x_j)=y_j$ that have both inputs and outputs involved in certain collisions, i.e., $\varphi_1^{in}(K_j,x_j)=\gamma_1^{in}(K_j,u_j)$ and $\varphi_3^{in}(K_j,w_j)=\gamma_3^{in}(K_j,y_j)$, the probability of which can be proven negligible. This slightly oversimplifies, and we refer to [28] for details.

The question is: how the *decisional inv-free problem* affects in this case? The point is that: the above strategy only works for encipherments that are inv-free in both 1st and 3rd rounds. When we query $\mathrm{P}(i_2,\delta_2,z_2)\to z_2'$, S may define z_2' such that many of the involved $\Pi_3^{\mathrm{P}}(K_j,w_j)$ become non-inv-free. It seems cumbersome to argue that there remain many (useful) inv-free $\Pi_3^{\mathrm{P}}(K_j,w_j)$.

Such simulator strategies are prohibited by our case condition. In detail, if S want to define z_2' such that $\Pi_3^{\mathrm{P}}(K_j,w_j)$ is non-inv-free for some j, S must find an ideal cipher input/output $\mathbf{IC}(K_j,x_j)=y_j$ such that $(\Pi_3^{-1})^{\mathrm{P}}(K_j,y_j)$ is non-inv-free (otherwise, there appears inconsistency). Though,

- Since it must satisfy $\varphi_1^{in}(K_j,x_j)=\gamma_1^{in}(K_j,u_j)$ ($\Pi_1(K_j,x_j)$ is also inv-free), it cannot be due to a backward query $\mathbf{IC}^{-1}(K_j,y_j)\to x_j$;
- Since only $o(2^n/\mathrm{poly}(n))$ $(\Pi_3^{-1})^{\mathcal{P}}(K_j,y)$ are non-inv-free for the involved key K_j, it cannot be due to a forward query $\mathbf{IC}(K_j,x_j)\to y_j$ either.

Thus, our attack strategy will reach $(\Pi_3^{-1})^{\mathrm{P}}(K_j,y_j)$ non-inv-free and succeed.

In our formal elaborations, this case is actually Subcase 3.1. We refer to the full version [28, Sect. 9.1] for details.

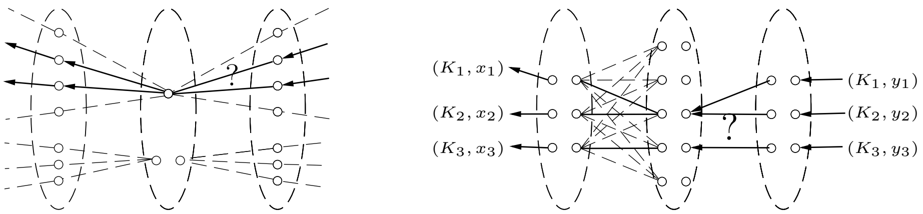

Fig. 6. Query structures used in attacking 3-round general ciphers. (Left) Structures for Case 1. The dashed lines show two examples of "stars": the top "star" centers around an inv-free 2nd round encipherment, while the bottom "star" centers around a non-inv-free 2nd round. (Right) Structures for Case 2. The dashes lines show a simple example of biclique $K_{3,5}$ (we certainly cannot draw "exponential-size"). The bold lines indicate the encipherments sampled by our attack, the arrows indicate the direction of our attack's evaluations, and the ? indicates where our attack checks equalities.

For the remaining, we first focus on the case that there are $\Theta(2^\kappa)$ keys K **s.t.** $\Omega(2^n/\text{poly}(n))$ $\Pi_1^{\mathcal{P}}(K,x)$ are non-inv-free, and that $\Omega(2^n/\text{poly}(n))$ $(\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free. Depending on whether the $\Omega(2^n/\text{poly}(n))$ 1st round outputs u have $\Pi_2^{\mathcal{P}}(K,u)$ non-inv-free or not, we further distinguish Case 2 and 3.

Case 2: there are $\boldsymbol{\Theta(2^\kappa)}$ keys $\boldsymbol{K}$ s.t. $\boldsymbol{\Omega\big(\frac{2^n}{\text{poly}(n)}\big)}$ $\boldsymbol{u}$ have $\boldsymbol{(\Pi_1^{-1})^{\mathcal{P}}(K,u)}$ non-inv-free and $\boldsymbol{\Pi_2^{\mathcal{P}}(K,u)}$ non-inv-free, and $\boldsymbol{\Omega\big(\frac{2^n}{\text{poly}(n)}\big)}$ $\boldsymbol{(\Pi_3^{-1})^{\mathcal{P}}(K,y)}$ are non-inv-free. A crucial example is the 3-round IEM cipher $\text{IEM3}(K,x) := K\oplus\mathbf{P}_3\big(K\oplus\mathbf{P}_2\big(K\oplus\mathbf{P}_1(K\oplus x)\big)\big)$. Let $u=\mathbf{P}_1(K\oplus x)$ in IEM3. Let's see an attack for intuition. We begin with three intermediate values $(K_1,u_1),(K_2,u_2),(K_3,u_3)$ that have $K_1\oplus u_1 = K_2\oplus u_2 \neq K_3\oplus u_3$, and then query P_1^{-1}, compute the plaintexts $x_1\leftarrow K_1\oplus\text{P}_1^{-1}(u_1)$, $x_2\leftarrow K_2\oplus\text{P}_1^{-1}(u_2)$ and $x_3\leftarrow K_3\oplus\text{P}_1^{-1}(u_3)$, and acquire the ciphertexts $\text{E}(K_1,x_1)\to y_1$, $\text{E}(K_2,x_2)\to y_2$ and $\text{E}(K_3,x_3)\to y_3$. With these, if we query $\text{P}_3^{-1}(K_1\oplus y_1)\to w$ and $\text{P}_3^{-1}(K_2\oplus y_2)\to w'$, then the simulator S shall define them such that $w\oplus K_1 = w'\oplus K_2$; if we query $\text{P}_3^{-1}(K_1\oplus y_1)\to w$ and $\text{P}_3^{-1}(K_3\oplus y_3)\to w'$, then S shall define $w\oplus K_1\neq w'\oplus K_3$. S cannot know our choice and thus won't be prepared correctly.

To translate this attack to the general 3-round model, we need to grasp its core idea. It turns out to be a structure of exponential size: if we view the range of P_1 and the domain of P_2 as two shores and the pairs (K,u) as edges, then we can build a biclique $K_{3,2^n}$. Due to this, given the P_1^{-1} and P_3^{-1} queries, there remain exponential possibilities for the three relevant encipherments $(K_1,x_1),(K_2,x_2)$ and (K_3,x_3), and S cannot pinpoint them. Furthermore, S does not know our choice of P_3^{-1}-queries either. These ideas were also used by Andreeva et al.'s attack on IEM3 [1, Sect. 3.3, full version] (though details slightly deviate).

In the general 3-round cipher, we should view possible inputs to $\mathcal{P}$ as shores and intermediate pairs (K,u) as edges to build a bipartite graph $\mathcal{G}$ (which resembles our previous treatments of 2-iteration), as shown in Fig. 6 (right). Again we need to prove that there indeed exists a biclique $K_{3,2^n}$ as a sub-graph in $\mathcal{G}$, and we resort to *Zarankiewicz numbers* [24]. Concretely, by Kővári, Sós and Turán (KST) [38] (restated in Sect. 2, Proposition 2), as long as κ is large enough (though still $\Theta(n)$), the number of edges is large enough and $K_{3,2^n}$ is guaranteed to exist. This enables finding and exploiting the three encipherments.

Regarding the *decisional inv-free problem*, the setting is simpler than Case 1. In detail, it can be proven that we can sample encipherments $(K_1,x_1),(K_2,x_2)$ and (K_3,x_3) that were unlikely queried by the simulator S. By this and by the case condition, we reach $(\Pi_3^{-1})^{\mathcal{P}}(K_1,y_1),(\Pi_3^{-1})^{\mathcal{P}}(K_2,y_2)$ and $(\Pi_3^{-1})^{\mathcal{P}}(K_3,y_3)$ with non-negligible probability $\Omega(1/\text{poly}(n))$ after querying $\text{E}(K_1,x_1)\to y_1$, $\text{E}(K_2,x_2)\to y_2$ and $\text{E}(K_3,x_3)\to y_3$. This fits into our expectations.

There remain subtleties: similarly to the 2-round case (Sect. 3.3), KST's result [38] only applies to simple graphs. When $\mathcal{G}$ is a multigraph with high multiplicity, we have to resort to a dedicated treatment. Interestingly, using the fact that there can be many edges between a single pair of vertexes, we are able to find three encipherments $(K_1,u_1),(K_2,u_2)$ and (K_3,u_3) that are similar to the above "simple" case. The involved structure is given in Fig. 7 (left).

We refer to the full version [28, Sect. 9.2 and 9.3] (Subcase 3.2) for details.

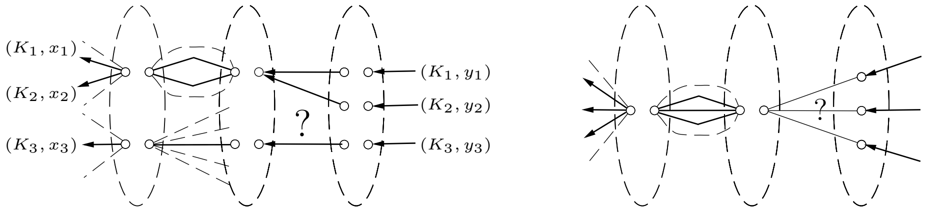

Fig. 7. Query structures used in attacking 3-round general ciphers, when the involved graphs contain heavy multi-edges. (Left) Structures for Case 2. The idea is adapted from Fig. 6 (right). The dashed arcs indicate that there are many (superpolynomial) distinct encipherments "crossing" the same pair of inputs in 1st and 2nd rounds. The dashed lines show that the number of possible encipherments "crossing" the two relevant $\mathcal{P}$-inputs are exponential. (Right) Structures for Case 4. The idea is adapted from Fig. 8 (right): we can find λ useful encipherments within a single pair of $\mathcal{P}$-inputs (in the 1st and 2nd rounds). The use of bold lines, arrows and ? follows Fig. 6.

Case 3: there are $\Theta(2^\kappa)$ keys K s.t. $\Omega(\frac{2^n}{\text{poly}(n)})$ u have $(\Pi_1^{-1})^{\mathcal{P}}(K,u)$ non-inv-free and $\Pi_2^{\mathcal{P}}(K,u)$ inv-free, and $\Omega(\frac{2^n}{\text{poly}(n)})$ $(\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free. Our attack in this case reuses already discussed ideas. In detail, (roughly) we sample a pair (K,u) from the $\Omega(2^{\kappa+n}/\text{poly}(n))$ pairs that have 1st round non-inv-free while 2nd round inv-free. We then evaluate backward to $x \leftarrow (\Pi_1^{-1})^{\mathrm{P}}(K,u)$, "wrap" by querying $\mathrm{E}(K,x) \rightarrow y$ and further $w \leftarrow (\Pi_3^{-1})^{\mathrm{P}}(K,y)$. Since $\Omega(2^n/\text{poly}(n))$ $(\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free, we reach $(\Pi_3^{-1})^{\mathrm{P}}(K,y)$ non-inv-free with a non-negligible probability and overcome the *decisional inv-free problem*, as shown in Fig. 8 (left). Since $\Pi_2(K,u)$ is inv-free, if we are interacting with the general 3-round cipher then it holds $\varphi_2^{in}(K,u) = \gamma_2^{in}(K,w)$ (as discussed in Sect. 3.2). On the other hand, if we are interacting with the ideal world $(\mathbf{IC}, S^{\mathbf{IC}})$, the simulator S only gains two P-calls $\mathrm{P}(\gamma_1^{in}(K,u))$ and $\mathrm{P}(\gamma_3^{in}(K,y))$. As discussed in Case 2, they won't enable S to pinpoint the encipherment (K,x). Consequently, S is unable to define simulated P and enforce the equality $\varphi_2^{in}(K,u) = \gamma_2^{in}(K,w)$.

In our formal elaborations [28, Sect. 9.4] (which use pigeonhole principle and non-degeneracy of φ_2^{in}), this corresponds to Subcase 3.3.

We then focus on the case that there are $\Theta(2^\kappa)$ keys K **s.t.** $\Omega(2^n/\text{poly}(n))$ $\Pi_1^{\mathcal{P}}(K,x)$ are non-inv-free, and that $o(2^n/\text{poly}(n))$ $(\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free. Similarly to Case 2 and 3, we further distinguish Case 4 and 5.

Case 4: there are $\Theta(2^\kappa)$ keys K s.t. $\Omega(\frac{2^n}{\text{poly}(n)})$ u have $(\Pi_1^{-1})^{\mathcal{P}}(K,u)$ non-inv-free and $\Pi_2^{\mathcal{P}}(K,u)$ non-inv-free, and $o(\frac{2^n}{\text{poly}(n)})$ $(\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free. In this case, we reuse the query structures found in Case 2. We also reuse the idea that the inv-free 3rd round allows checking consistency.

In detail, consider the bipartite graph $\mathcal{G}$ built between the 1st and 2nd round (which resembles Case 2). When $\mathcal{G}$ is (roughly) simple, we can find a biclique $K_{\lambda,2^n}$ with $\lambda = m_{max}$, which resembles Case 2. We then sample one

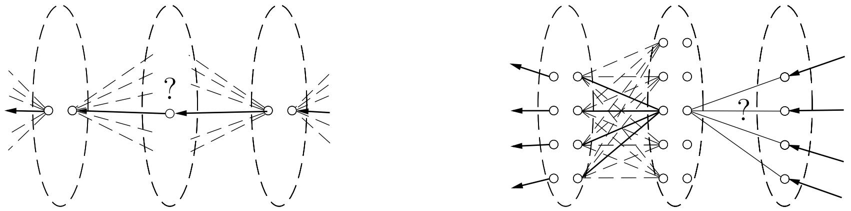

Fig. 8. Query structures used in attacking 3-round general ciphers. (Left) Structures for Case 3. The dashed lines show that the number of possible encipherments "crossing" the two relevant $\mathcal{P}$-inputs are exponential (though, they may not overlap). (Right) Structures for Case 4. The figure shows a simple example with $\lambda = 4$. The use of bold lines, arrows and ? follows Fig. 6.

vertex (i_2, δ_2, z_2) from the right shore of $K_{\lambda,2^n}$, pinpointing λ encipherments $(K_1, u_1), ..., (K_\lambda, u_\lambda)$ that invoke $\mathrm{P}(i_2, \delta_2, z_2)$ in the 2nd round. See Fig. 8 (left). We then evaluate backward $x_1 \leftarrow (\mathit{\Pi}_1^{-1})^{\mathrm{P}}(K_1, u_1), ..., x_\lambda \leftarrow (\mathit{\Pi}_1^{-1})^{\mathrm{P}}(K_\lambda, u_\lambda)$, "wrap" $\mathrm{E}(K_1, x_1) \to y_1, ..., \mathrm{E}(K_\lambda, x_\lambda) \to y_\lambda$. As only $o\big(\frac{2^n}{\mathrm{poly}(n)}\big)$ $(\mathit{\Pi}_3^{-1})^{\mathcal{P}}(K, y)$ are non-inv-free, we likely reach $(\mathit{\Pi}_3^{-1})^{\mathrm{P}}(K_1, y_1), ..., (\mathit{\Pi}_3^{-1})^{\mathrm{P}}(K_\lambda, y_\lambda)$ inv-free and overcome the *decisional inv-free problem*, as shown in Fig. 8 (right).

In the real world, the 2nd round outputs $(K_1, w_1), ..., (K_\lambda, w_\lambda)$ of these encipherments are derivable from $(K_1, w_1), ..., (K_\lambda, w_\lambda)$ using a fixed $z_2' \in \{0,1\}^{m(i_2)}$. Meanwhile, they have $\varphi_3^{in}(K_1, w_1) = \gamma_3^{in}(K_1, y_1), ..., \varphi_3^{in}(K_\lambda, w_\lambda) = \gamma_3^{in}(K_\lambda, y_\lambda)$. To simulate consistently, the simulator S in the ideal world has to find a corresponding $z_2' \in \{0,1\}^{m(i_2)}$ satisfying the λ equalities for the ideal cipher responses $y_1, ..., y_\lambda$. Since $\lambda = m_{max}$, this can be proven infeasible.

When $\mathcal{G}$ is a multigraph with high multiplicity, a single pair of vertexes already suffices to pinpoint λ encipherments $(K_1, u_1), ..., (K_\lambda, u_\lambda)$ that invoke the same $\mathrm{P}(i_2, \delta_2, z_2)$ in the 2nd round, as shown in Fig. 7 (right). Our above idea thus remains applicable.

In our formal elaborations [28, Sect. 9.5], this corresponds to Subcase 3.4.

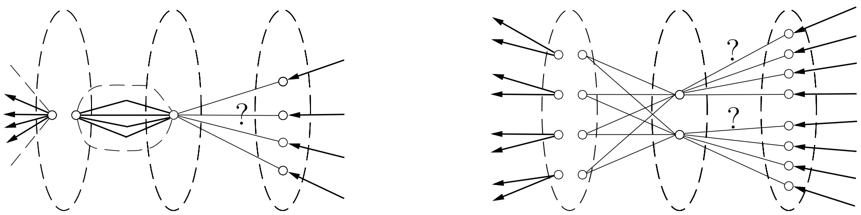

Fig. 9. Query structures used in attacking 3-round general ciphers, Case 5. (Left) When the graph contains heavy multi-edges, we can find λ useful encipherments within a single pair of $\mathcal{P}$-inputs (in the 1st and 2nd rounds). The figure shows a simple example with $\lambda = 4$. (Right) When the graph does not contain too many multi-edges (and the biclique $K_{\lambda,2}$ exists).

Case 5: there are $\boldsymbol{\Theta(2^\kappa)}$ keys $\boldsymbol{K}$ s.t. $\boldsymbol{\Omega(\frac{2^n}{\mathrm{poly}(n)})}$ $\boldsymbol{u}$ have $\boldsymbol{(\Pi_1^{-1})^{\mathcal{P}}(K,u)}$ non-inv-free and $\boldsymbol{\Pi_2^{\mathcal{P}}(K,u)}$ inv-free, and $\boldsymbol{o(\frac{2^n}{\mathrm{poly}(n)})}$ $\boldsymbol{(\Pi_3^{-1})^{\mathcal{P}}(K,y)}$ are non-inv-free. Again, consider the bipartite graph $\mathcal{G}$ built between the 1st and 2nd round (which resembles Cases 2 and 4). When $\mathcal{G}$ is a multigraph with high multiplicity, we reuse the idea of Case 4 and exploit the structure shown in Fig. 9 (left). The case that $\mathcal{G}$ is (roughly) simple turns out to be the most complicated, and many of our earlier attempts failed. Our eventual idea is built upon a polynomial-size boomerang structure in the 1st and 2nd rounds, which is depicted in Fig. 9 (right).

In detail, we seek for a biclique $K_{\lambda,2}$, $\lambda = O(m_{max})$, in $\mathcal{G}$, as shown in Fig. 9 (right). Again by KST [38], such bicliques exist as long as κ is large enough (though still $\Theta(m_{max}n) = \Theta(\mathrm{poly}(n))$).

The biclique $K_{\lambda,2}$ pinpoints two groups of encipherments, with each group colliding on a 2nd round $\mathcal{P}$-call, as shown in Fig. 9. Therefore, in the real world, there are two $\mathcal{P}$-outputs that are consistent with all the 2λ encipherments. Meanwhile, every encipherment in one group is paired with an encipherment in the other group, such that the two encipherments collide on the 1st round $\mathcal{P}$-call. By these, in the ideal world, the simulator S has to seek for 2λ ideal cipher queries that have both inputs and outputs involved in certain collisions. Namely, the 2λ ideal cipher queries can be arranged in a $2 \times \lambda$ matrix, such that:

- For every pair of ideal cipher queries in every column, the corresponding simulated encipherments collide on the 1st round $\mathcal{P}$-call; and
- For each group of λ ideal cipher queries in each row, there exists a response z_2' that satisfy certain relation with their λ ciphertexts.

When $\lambda = O(m_{max})$, this can be proven infeasible.

In our formal elaborations [28, Sect. 9.6], this corresponds to Subcase 3.5. Some of our earlier failed attempts are also available there.

Other cases: there are $\boldsymbol{\Theta(2^\kappa)}$ keys $\boldsymbol{K}$ s.t. $\boldsymbol{o(\frac{2^n}{\mathrm{poly}(n)})}$ $\boldsymbol{\Pi_1^{\mathcal{P}}(K,x)}$ are non-inv-free and $\boldsymbol{\Omega(\frac{2^n}{\mathrm{poly}(n)})}$ $\boldsymbol{(\Pi_3^{-1})^{\mathcal{P}}(K,y)}$ are non-inv-free. Then, if most $w = (\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are inv-free w.r.t. Π_2, it follows the above Case 4 by symmetry; if most $w = (\Pi_3^{-1})^{\mathcal{P}}(K,y)$ are non-inv-free w.r.t. Π_2, it follows the above Case 5 by symmetry. We thereby complete the case-study.

Again, we refer to [28, Sect. 9] for the complicated details.

4 Fundamental Properties

By the *notion* of blockciphers, a blockcipher shall be *deterministic* and *efficiently invertible*. The latter has been reflected in Fig. 10 (and Figs. 11 and 12 as well). Below we formalize the former for blockcipher oracle procedures.

Definition 1 (Deterministicness). *An oracle procedure $E^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ instantiating a blockcipher must be deterministic, meaning that for any*

$(K,x) \in \mathcal{K} \times \{0,1\}^n$*, let* $y = E^{\mathcal{P}}(K,x)$*. Then, the transcripts of* $\mathcal{P}$*-queries and responses obtained during encipherment* $E^{\mathcal{P}}(K,x)$ *and decipherment* $(E^{-1})^{\mathcal{P}}(K,y)$ *are always identical. (E.g., if* $E^{\mathcal{P}}(K,x)$ *queries* $\mathcal{P}(i,\delta,z) \to z'$ *at some stage, then* $(E^{-1})^{\mathcal{P}}(K,y)$ *queries either* $\mathcal{P}(i,\delta,z) \to z'$ *or* $\mathcal{P}(i,\overline{\delta},z') \to z$ *at some stage.)*

Two more properties/assumptions that we rely on are **oracle-independence of sub-procedures** and **non-degeneracy of** $E^{\mathcal{P}}$.

Oracle-independence means sub-procedures in $E^{\mathcal{P}}$ must be *oracle-independent*. Since the oracle procedure $E^{\mathcal{P}}$ (or black-box cryptographic construction) has a fixed description, this seems obvious (and indeed common in black-box constructions [25] and impossibility proofs [9,44]). Though, we highlight it for clarity. Interestingly, ad hoc blockciphers also strive for such independence (probably to avoid unexpected internal dependency). For example, in AES, the ShiftRows and MixColumns steps are rather independent from SubBytes.

Non-degeneracy means no encipherment $E^{\mathcal{P}}(K,x)$ can be approximately computed using less $\mathcal{P}$ calls than $E^{\mathcal{P}}$, i.e., $E^{\mathcal{P}}$ cannot be "simplified". Formally,

Definition 2 ((Everywhere) Non-degenerate Oracle Procedure). *An oracle procedure* $E^{\mathcal{P}}$ *is (everywhere)* $\varepsilon_{de(E)}$*-non-degenerate, if*

$$\max_{E',K,x}\Big\{\Pr_{\mathcal{P}}\big[(E')^{\mathcal{P}}(K,x) = E^{\mathcal{P}}(K,x)\big]\Big\} \leq \varepsilon_{de(E)} = negl(n). \qquad (3)$$

where the maximum is taken over all $(K,x) \in \{0,1\}^{\kappa} \times \{0,1\}^n$ *and all oracle procedures* $(E')^{\mathcal{P}}$ *such that the number of* $\mathcal{P}$*-calls made during computing* $(E')^{\mathcal{P}}(K,x)$ *is less than* $E^{\mathcal{P}}(K,x)$*.*

Why non-degenerate? If $E1^{\mathcal{P}}$ is $1/\text{poly}(n)$-non-degenerate, then there is an obvious differentiator with advantage $1/\text{poly}(n) - 2^{-n}$. More importantly, a t-call blockcipher "uses" all of its t $\mathcal{P}$-calls "effectively" only if it is non-degenerate.

5 General 1-Call Blockciphers

We first elaborate on our 1-call cipher model $E1^{\mathcal{P}}$ in Sect. 5.1. Then, we characterize the properties of $E1^{\mathcal{P}}$ in Sect. 5.2. Our formal conclusion on $E1$ insecurity is given in Sect. 5.3.

5.1 General Model of 1-call Blockciphers/Rounds

We consider any blockcipher oracle procedure $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ that is built from the permutation family $\mathcal{P}$ in the following way. Let φ^{in} and φ^{out} be two arbitrary deterministic functions that are computable by the computational class of the differentiator(s). Then, $E1^{\mathcal{P}}(K,x) := \varphi^{out}\big(K, \mathcal{P}\big(\varphi^{in}(K,x)\big), x\big)$.

Since blockciphers are *efficiently invertible* by definitions, $E1^{\mathcal{P}}$ is accomplished by $(E1^{-1})^{\mathcal{P}}(K,y) := \gamma^{out}\big(K, \mathcal{P}\big(\gamma^{in}(K,y)\big), y\big)$ using two other deterministic functions γ^{in} and γ^{out}. We stress that to ensure $(E1^{-1})^{\mathcal{P}}\big(K, E1^{\mathcal{P}}(K,x)\big) \equiv x$, γ^{in} and γ^{out} are strongly correlated with φ^{in} and φ^{out}, and this will be crucial for our attack. A formal description using pseudocode is given in Fig. 10.

```
Algorithm $E1^{\mathcal{P}}(K,x)$ // $(K,x) \in \mathcal{K} \times \{0,1\}^n$
    $(i,\delta,z) \leftarrow \varphi^{in}(K,x)$
    $z' \leftarrow \mathcal{P}(i,\delta,z)$
    $y \leftarrow \varphi^{out}(K,z',x)$
    return $y$
```

```
Algorithm $(E1^{-1})^{\mathcal{P}}(K,x)$
    $(i,\delta,z) \leftarrow \gamma^{in}(K,y)$
    $z' \leftarrow \mathcal{P}(i,\delta,z)$
    $x \leftarrow \gamma^{out}(K,z',y)$
    return $x$
```

Fig. 10. Definition of the 1-call blockcipher $E1^{\mathcal{P}}$. $\varphi^{in}, \varphi^{out}, \gamma^{in}$, and γ^{out} are all deterministic and oracle-independent.

Examples to facilitate understanding. First, the key-prepended Feistel round [16] uses $\mathbf{F} : \{0,1\}^{\kappa+n/2} \to \{0,1\}^{n/2}$ and defines $\Psi^{\mathbf{F}}(K,x) := \mathsf{right}_{n/2}(x) \oplus \mathbf{F}(K \| \mathsf{left}_{n/2}(x)) \| \mathsf{left}_{n/2}(x)$. It is an $E1$ instance with

$$\begin{aligned} \varphi^{in}(K,x) &:= \big(i, +, K \| \mathsf{right}_{n/2}(x) \| [0]_{n/2}\big), \\ \varphi^{out}(K,z',x) &:= \mathsf{right}_{n/2}(x) \oplus \mathsf{right}_{n/2}(z') \| \mathsf{left}_{n/2}(x) \end{aligned} \tag{4}$$

using an index i with $m(i) = \kappa + n$ (and truncated permutation [14]).

Second, the IEM round [23] defines $\mathrm{EM}^{\mathbf{P}}(K,x) := K \oplus \mathbf{P}(K \oplus x)$ for $\mathbf{P} \in \mathsf{Perm}(n)$. It is an $E1$ instance with

$$\varphi^{in}(K,x) := (i, +, K \oplus x),\ \ \varphi^{out}(K,z',x) := K \oplus z' \tag{5}$$

using an index i with $m(i) = n$.

Finally, a "key-alternating" Misty-R cipher round [2, Chapter 3.18.8] defines $\text{Misty-R}^{\mathbf{P}}(K,x) := \mathbf{P}\big(K \oplus \mathsf{right}_{n/2}(x)\big) \big\| \big(\mathsf{left}_{n/2}(x) \oplus \mathbf{P}\big(K \oplus \mathsf{right}_{n/2}(x)\big)\big)$ for $\mathbf{P} \in \mathsf{Perm}(n/2)$. It is an $E1$ instance with

$$\varphi^{in}(K,x) := \big(i, +, K \oplus \mathsf{right}_{n/2}(x)\big),\ \ \varphi^{out}(K,z',x) := z' \big\| \big(\mathsf{left}_{n/2}(x) \oplus z'\big) \tag{6}$$

using an index i with $m(i) = n/2$.

It is easy to see unbalanced Feistel [2, Chapter 1.3.1], Lai-Massey [2, Chapter 1.5] and keyed Feistel rounds are instances of $E1$ as well. Though, $E1$ does not cover multi-line generalized Feistel [2, Chapter 1.3.1] (which makes multiple $\mathcal{P}$ calls per round) and Swap-Or-Not [29] (which uses small-range functions).

5.2 Properties of 1-call Blockciphers/Rounds

We first introduce several helper sets. We then discuss properties of *data-dependent encipherments*, *inverse-free* and *non-inverse-free encipherments* in turn.

Notations. For any 1-call cipher $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ and any K in its keyspace $\mathcal{K}$, define

$$\begin{aligned} \mathsf{Dom}_{\mathsf{if}}(E1,K) := \Big\{ & x \in \{0,1\}^n : \delta = \delta',\ \text{where } (i,\delta,z) = \varphi^{in}(K,x), \\ & (i,\delta',z') = \gamma^{in}(K,y),\ y = E1^{\mathcal{P}}(K,x) \Big\}, \\ \mathsf{Rng}_{\mathsf{if}}(E1,K) := \Big\{ & y \in \{0,1\}^n : \delta = \delta',\ \text{where } (i,\delta,z) = \gamma^{in}(K,y), \\ & (i,\delta',z') = \varphi^{in}(K,x),\ x = (E1^{-1})^{\mathcal{P}}(K,y) \Big\}, \end{aligned}$$

$$\mathsf{Dom}_{\mathsf{ni}}(E1,K) := \{0,1\}^n \backslash \mathsf{Dom}_{\mathsf{if}}(E1,K),\ \ \mathsf{Rng}_{\mathsf{ni}}(E1,K) := \{0,1\}^n \backslash \mathsf{Rng}_{\mathsf{if}}(E1,K). \tag{7}$$

For $x \in \mathsf{Dom}_{\mathsf{if}}(E1, K)$, the encipherment $E1^{\mathcal{P}}(K, x)$ and the corresponding decipherment $(E1^{-1})^{\mathcal{P}}(K, y)$ call $\mathcal{P}$ on the same direction. Therefore, $E1^{\mathcal{P}}(K, x)$ is *inverse-free* (inv-free for short), as reflected by the subscript if. Otherwise, $E1^{\mathcal{P}}(K, x)$ is *non-inverse-free* (non-inv-free), as reflected by ni. We remark that $E1^{\mathcal{P}}(K, x) = E'(\mathcal{P}(f(K)), x)$ is also inv-free, although it may not match classical understandings.

For $\mathsf{tag} \in \{\mathsf{ni}, \mathsf{if}\}$, define sets for plaintexts/ciphertexts in $\mathsf{Dom}_{\mathsf{tag}}/\mathsf{Rng}_{\mathsf{tag}}$ that are mapped to a certain $\mathcal{P}$ input (i, δ, z):

$$\begin{aligned}
&\mathsf{Dom}_{\mathsf{tag}}(E1, K, i, \delta, z) := \big\{x \in \mathsf{Dom}_{\mathsf{tag}}(E1, K) : \varphi^{in}(K, x) = (i, \delta, z)\big\},\\
&\mathsf{Rng}_{\mathsf{tag}}(E1, K, i, \delta, z) := \big\{y \in \mathsf{Rng}_{\mathsf{tag}}(E1, K) : \gamma^{in}(K, y) = (i, \delta, z)\big\}. \qquad (8)\\
&\mathsf{Dom}_{\mathsf{tag}}(E1, K, i, \delta) := \cup_{z \in \{0,1\}^{m(i)}} \mathsf{Dom}_{\mathsf{tag}}(E1, K, i, \delta, z),\\
&\mathsf{Rng}_{\mathsf{tag}}(E1, K, i, \delta) := \cup_{z \in \{0,1\}^{m(i)}} \mathsf{Rng}_{\mathsf{tag}}(E1, K, i, \delta, z). \qquad (9)
\end{aligned}$$

We slightly abuse the notation $\mathsf{Rng}_{\star}$ to denote the actual ranges of the input functions φ^{in} and γ^{in}. In detail, for $\mathsf{tag} \in \{\mathsf{ni}, \mathsf{if}\}$, define

$$\begin{aligned}
&\mathsf{Rng}_{\mathsf{tag}}(\varphi^{in}, K) := \big\{(i, \delta, z) : (i, \delta, z) = \varphi^{in}(K, x) \text{ for some } x \in \mathsf{Dom}_{\mathsf{tag}}(E1, K)\big\},\\
&\mathsf{Rng}_{\mathsf{tag}}(\gamma^{in}, K) := \big\{(i, \delta, z) : (i, \delta, z) = \gamma^{in}(K, y) \text{ for some } y \in \mathsf{Rng}_{\mathsf{tag}}(E1, K)\big\}.\\
&\mathsf{Rng}_{\mathsf{tag}}(\varphi^{in}) := \cup_{K \in \mathcal{K}} \mathsf{Rng}_{\mathsf{tag}}(\varphi^{in}, K), \quad \mathsf{Rng}_{\mathsf{tag}}(\gamma^{in}) := \cup_{K \in \mathcal{K}} \mathsf{Rng}_{\mathsf{tag}}(\gamma^{in}, K). \qquad (10)
\end{aligned}$$

On data-dependence. As indicated by the partition $\{0,1\}^n = \mathsf{Dom}_{\mathsf{if}}(E1, K) \sqcup \mathsf{Dom}_{\mathsf{ni}}(E1, K)$, the inv-freeness of $E1^{\mathcal{P}}(K, x)$ can be data-dependent. Though, as mentioned in Sect. 3.1, (surprisingly) *one can decide whether an encipherment $E1^{\mathcal{P}}(K, x)$ is inv-free without querying $\mathcal{P}$*. This turns out crucial in our attacks.

Lemma 1 (Inv-freeness is oracle-independent). *Consider the blockcipher $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ in Fig. 10. Then, for any pair $(K, x) \in \mathcal{K} \times \{0,1\}^n$, resp. $(K, y) \in \mathcal{K} \times \{0,1\}^n$, whether $x \in \mathsf{Dom}_{\mathsf{if}}(E1, K)$, resp. $y \in \mathsf{Rng}_{\mathsf{if}}(E1, K)$, can be determined without querying $\mathcal{P}$.*

Proof. Assume otherwise, and let (K, x) be the input such that whether $x \in \mathsf{Dom}_{\mathsf{if}}(E1, K)$ depends on $\mathcal{P}$. Let $(i, \delta, z) = \varphi^{in}(K, x)$ and $y = E1^{\mathcal{P}}(K, x)$. Since $E1^{\mathcal{P}}(K, x)$ only makes one query $\mathcal{P}(i, \delta, z)$ to $\mathcal{P}$, $E1^{\mathcal{P}}(K, x)$ is inv-free if and only if $\mathcal{P}(i, \delta, z)$ is in a certain subset of $\{0,1\}^{m(i)}$. Namely, there exists a partition $\{0,1\}^{m(i)} = \mathcal{Z}_{\delta} \cup \mathcal{Z}_{\overline{\delta}}$ such that $x \in \mathsf{Dom}_{\mathsf{if}}(E1, K)$ if and only if $\mathcal{P}(i, \delta, z) \in \mathcal{Z}_{\delta}$.

However, let $(i, \delta', z') = \gamma^{in}(K, y)$, then $y \in \mathsf{Rng}_{\mathsf{if}}(E1, K)$ if and only if $\delta' = \delta$. This means it always holds $\mathcal{P}(i, \delta, z) \in \mathcal{Z}_{\delta'}$, where δ' is *fixed* by the definition of the function γ^{in}. This violates our assumption that γ^{in} is oracle-independent.

Therefore, one can decide if $x \in \mathsf{Dom}_{\mathsf{if}}(E1, K)$ solely by computations. The argument for $y \in \mathsf{Rng}_{\mathsf{if}}(E1, K)$ is similar by symmetry. □

Properties of inv-free encipherments. We now formalize the intuitive weaknesses of inv-free encipherments discussed in Sect. 3.1.

Lemma 2 (Inv-freeness preserves partial inputs). *Consider the 1-call blockcipher $E1^{\mathcal{P}}$ in Fig. 10. Then, for any pair (K,x), $x \in \mathsf{Dom}_{\mathsf{if}}(E1,K)$, it holds $\gamma^{in}(K,y)=\varphi^{in}(K,x)$ for $y=E1^{\mathcal{P}}(K,x)$. It further implies $|\mathsf{Dom}_{\mathsf{if}}(E1,i,\delta,z)| = |\mathsf{Rng}_{\mathsf{if}}(E1,i,\delta,z)|$ for any $i \in \mathcal{I}$, $\delta \in \{+,-\}$ and $z \in \{0,1\}^{m(i)}$. Proof: this is a straightforward implication of Fig. 10 and Definition 1.*

The second observation follows by non-degeneracy: if there exists one inv-free encipherment $E1^{\mathcal{P}}(K,x)$, then there must exist superpolynomially many.

Lemma 3 (Inv-freeness can't be unique). *Consider the 1-call blockcipher $E1^{\mathcal{P}}$ in Fig. 10. If $E1^{\mathcal{P}}$ is $\varepsilon_{de(E1)}$-non-degenerate in the sense of Definition 2, then for any (K,i,δ,z) such that $\mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)\neq\emptyset$, it holds*

$$\big|\mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)\big| > 1/\varepsilon_{de(E1)} = \Omega(poly(n)). \tag{11}$$

Proof. Assume otherwise, i.e., $|\mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)| \le \varepsilon_{de(E1)}^{-1}$ for some (K,i,δ,z). By Lemma 2, for any $x \in \mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)$, the corresponding ciphertext $y = E1^{\mathcal{P}}(K,x)$ must have $y \in \mathsf{Rng}_{\mathsf{if}}(E1,K,i,\delta,z)$. By Lemma 2, $|\mathsf{Rng}_{\mathsf{if}}(E1,K,i,\delta,z)| = |\mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)| \le 1/\varepsilon_{de(E1)}$. By these, for any $x \in \mathsf{Dom}_{\mathsf{if}}(E1,K,i,\delta,z)$, one can uniformly pick $y \xleftarrow{\$} \mathsf{Rng}_{\mathsf{if}}(E1,K,i,\delta,z)$ to *encipher* (K,x) *without querying* $\mathcal{P}$ *at all*, and the success probability is at least $\varepsilon_{de(E1)}$. This contradicts the assumption that $E1^{\mathcal{P}}$ is $\varepsilon_{de(E1)}$-non-degenerate as Eq. (3). □

An implication of Lemma 3 is that the ranges of φ^{in} and γ^{in} cannot be too large. Due to page limits, the proof is deferred to the full version [28].

Lemma 4 (Functions in inv-free encipherments). *Consider the 1-call blockcipher $E1^{\mathcal{P}}$ in Fig. 10. If $E1^{\mathcal{P}}$ is $\varepsilon_{de(E1)}$-non-degenerate (see Definition 2), then it holds $\big|\mathsf{Rng}_{\mathsf{if}}(\varphi^{in},K)\big| = \big|\mathsf{Rng}_{\mathsf{if}}(\gamma^{in},K)\big| \le |\mathsf{Dom}_{\mathsf{if}}(E1,K)|\cdot\varepsilon_{de(E1)} \le 2^n\cdot\varepsilon_{de(E1)}$.*

Properties of non-inv-free encipherment. For $E1^{\mathcal{P}}(x)$, $x \in \mathsf{Dom}_{\mathsf{ni}}(E1,K)$, our first observation is that $E1^{\mathcal{P}}(x)$ cannot query *wide random permutations* (as discussed in Sect. 3.2). We now elaborate on the "regularity" of φ^{in} and γ^{in}. For example, in the IEM round (see Eq. (5)), for every $K \in \{0,1\}^n$ we have $|\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+)| = 2^n = 2^{m(i)}$, and for every $z \in \{0,1\}^n$ we have $|\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+,z)| = 1 = |\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+)|/2^{m(i)}$. In the "key-alternating" Misty-R round (see Eq. (5)), we have $|\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+)| = 2^n$ for $K \in \{0,1\}^{n/2}$ and $|\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+,z)| = 2^n = |\mathsf{Dom}_{\mathsf{ni}}(E1,K,i,+)|/2^{m(i)}$ with $m(i) = n/2$ for every $z \in \{0,1\}^{n/2}$.

Below we formalize the above first idea and show that the actual ranges of the functions φ^{in} and γ^{in} must be somewhat limited.

Lemma 5 (Non-inv-free encipherments cannot query wide P). *Consider the 1-call blockcipher $E1^{\mathcal{P}}$ in Fig. 10. Then:*

- *For any key $K \in \mathcal{K}$ and any $x \in \mathsf{Dom}_{\mathsf{ni}}(E1,K)$, let $(i,\delta,z)=\varphi^{in}(K,x)$, then it holds $i \in \mathcal{I}_{\le n}$;*

– *Similarly, for any key $K \in \mathcal{K}$ and any $y \in \mathsf{Rng}_{\mathsf{ni}}(E1, K)$, let $(i, \delta, z) = \gamma^{in}(K, y)$, then it holds $i \in \mathcal{I}_{\leq n}$.*

Consequently, $\left|\mathsf{Rng}_{\mathsf{ni}}(\varphi^{in})\right| \leq |\mathcal{I}_{\leq n}|2^{n+1}$, $\left|\mathsf{Rng}_{\mathsf{ni}}(\gamma^{in})\right| \leq |\mathcal{I}_{\leq n}|2^{n+1}$.

Proof. Assume otherwise, then there exists (K, x) such that $x \in \mathsf{Dom}_{\mathsf{ni}}(E1, K)$, and $(i, \delta, z) = \varphi^{in}(K, x)$ has $i \in \mathcal{I}_{>n}$. This means $|z| = m(i) > n$.

Furthermore, the oracle response $\mathcal{P}(\varphi^{in}(K, x)) = z'$ must be that there exists $y \in \mathsf{Rng}_{\mathsf{ni}}(E1, K)$ such that $\gamma^{in}(K, y) = (i, \overline{\delta}, z')$. Since $x \in \mathsf{Dom}_{\mathsf{ni}}(E1, K) \subseteq \{0,1\}^n$ and $y \in \mathsf{Rng}_{\mathsf{ni}}(E1, K) \subseteq \{0,1\}^n$, the number t of z and z' related by such relation is at most 2^n, meaning that $\mathcal{P}(i, \cdot, \cdot)$ must map a set of $t \leq 2^n$ possible z values (that are determined by φ^{in}) to a set of $t \leq 2^n$ possible z' values (that are determined by γ^{in}). This violates the oracle-independence assumption on φ^{in} and γ^{in}. Therefore, for any (K, x), $x \in \mathsf{Dom}_{\mathsf{ni}}(E1, K)$, let $(i, \delta, z) = \varphi^{in}(K, x)$, then it holds $i \in \mathcal{I}_{\leq n}$, i.e., $m(i) \leq n$. It thus follows $\left|\mathsf{Rng}_{\mathsf{ni}}(\varphi^{in})\right| \leq \left|\{+, -\}\right| \times |\mathcal{I}_{\leq n}| \times 2^n \leq |\mathcal{I}_{\leq n}|2^{n+1}$ and $\left|\mathsf{Rng}_{\mathsf{ni}}(\gamma^{in})\right| \leq |\mathcal{I}_{\leq n}|2^{n+1}$. □

We then formalize the above (somewhat surprising) "regularity" idea (the proof is deferred to [28] due to page limits).

Lemma 6 (Regularity in non-inv-free encipherments). *Consider the 1-call blockcipher $E1^{\mathcal{P}}$ in Fig. 10. Then, for any $K \in \mathcal{K}$ and any $(i, \delta) \in \mathcal{I}_{\leq n} \times \{+, -\}$, the restriction of φ^{in} to $\mathsf{Dom}_{\mathsf{ni}}(E1, K, i, \delta)$ (resp., the restriction of γ^{in} to $\mathsf{Rng}_{\mathsf{ni}}(E1, K, i, \delta)$) is regular. I.e., the following holds for any $z, z' \in \{0,1\}^{m(i)}$*

$$\left|\mathsf{Dom}_{\mathsf{ni}}(E1, K, i, \delta, z)\right| = \left|\mathsf{Rng}_{\mathsf{ni}}(E1, K, i, \overline{\delta}, z')\right| = \frac{\left|\mathsf{Dom}_{\mathsf{ni}}(E1, K, i, \delta)\right|}{2^{m(i)}}.$$

This also means $\left|\mathsf{Dom}_{\mathsf{ni}}(E1, K, i, \delta)\right|$ must be divisible by $2^{m(i)}$.

By Lemma 5, we can derive the collision probability among images of φ^{in} and γ^{in} as follows (the proof is deferred to [28]).

Corollary 1 (Probability of collisions). *For any (i, δ, z) and any set $\mathcal{S} \subseteq \{0,1\}^n$ with $|\mathcal{S}| = poly(n)$, when n is sufficiently large it holds*

$$\Pr\left[y \xleftarrow{\$} \{0,1\}^n \backslash \mathcal{S} : \gamma^{in}(K, y) = (i, \delta, z) \;\middle|\; y \in \mathsf{Rng}_{\mathsf{ni}}(E1, K)\right] \leq \frac{2}{2^{m_{min}}}, \quad (12)$$

$$\Pr\left[x \xleftarrow{\$} \{0,1\}^n \backslash \mathcal{S} : \varphi^{in}(K, x) = (i, \delta, z) \;\middle|\; x \in \mathsf{Dom}_{\mathsf{ni}}(E1, K)\right] \leq \frac{2}{2^{m_{min}}}. \quad (13)$$

5.3 Attack 1-Call Blockciphers

After the preparations in Sect. 5.2, we are able to establish insecurity of 1-call ciphers $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ of Fig. 10.

Theorem 1 (Differentiability of $E1^{\mathcal{P}}$). *Let $E1^{\mathcal{P}} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher defined by Fig. 10. Assume that $E1^{\mathcal{P}}$ is deterministic and $\varepsilon_{de(E1)}$-non-degenerate in the sense of Definition 2, and its keyspace has $|\mathcal{K}| \geq 2|\mathcal{I}_{\leq n}| + 1 = O(poly(n))$. Then, when n is sufficiently large, there exists a differentiator $D1^{E,P}$ making at most $\lceil m_{max}/n \rceil + 2$ queries and has an advantage at least $1 - \frac{{m_{max}}^2}{2^n} - \frac{2}{2^{m_{min}}} = 1 - negl(n)$.*

It is crucial to restrict $|\mathcal{K}| > |\mathcal{I}_{\leq n}|$: otherwise, $\mathcal{P}$ may already offer $|\mathcal{K}|$ independent n-bit random permutations. There are two purposes to consider $|\mathcal{K}| = \mathrm{poly}(n)$. First, it strengthens the negative result (i.e., even indifferentiable cipher of logarithmic key length is impossible). Second, such $D1$ can function as subroutines of $D2$ and $D3$ in Sect. 6 and 7.

The full proof is available in [28].

6 Attack 2-Call Iterated Blockciphers

For 2- and 3-call ciphers, we restrict to *iterated blockciphers* that are built from *key derivation functions* and *rounds*. For a 2-call cipher $E2^{\mathcal{P}}$, this means encipherment $E2^{\mathcal{P}}(K,x)$ must proceed with either of the following flows:

- **Type-I**: $E2^{\mathcal{P}}(K,x) = \Pi_3^{\mathcal{P}}\big(K\|\mathsf{kd}^{\mathcal{P}}(K),x\big)$ for a 1-call function $\mathsf{kd}^{\mathcal{P}} : \{0,1\}^{\kappa} \to \{0,1\}^{m_{max}}$ and a 1-call cipher $\Pi_3^{\mathcal{P}} : \{0,1\}^{\kappa+m_{max}} \times \{0,1\}^n \to \{0,1\}^n$, or
- **Type-II**: $E2^{\mathcal{P}}(K,x) = \Pi_2^{\mathcal{P}}\big(K, \Pi_1^{\mathcal{P}}(K,x)\big)$ for two 1-call ciphers/rounds $\Pi_1^{\mathcal{P}}, \Pi_2^{\mathcal{P}} : \{0,1\}^{\kappa} \times \{0,1\}^n \to \{0,1\}^n$.

The keyspace is partitioned $\mathcal{K} = \mathcal{K}^{(0)} \sqcup \mathcal{K}^{(1)}$, such that $E2^{\mathcal{P}}(K,\cdot)$ follows **Type-I** encipherment if and only if $K \in \mathcal{K}^{(1)}$. Formally, we consider the cipher $E2^{\mathcal{P}}$ defined in Fig. 11. As mentioned in the Introduction, there is no need to use multiple $\mathcal{P}_1, \mathcal{P}_2, \ldots$, since $\mathcal{P}$ already provides multiple independent permutations.

As discussed the overview (Sect. 3.3), we make an additional non-degenerate assumption on the input functions φ^{in}. Formally,

Definition 3 (Non-degenerate Keyed Function). *A keyed function $\varphi^{in}(\cdot,\cdot)$ is $\varepsilon_{de(\varphi^{in})}$-non-degenerate, if the following two upper bounds hold (recall from Lemma 1 that the set $\mathsf{Dom}_{\mathsf{if}}(E1,K)$ is fully determined by φ^{in}):*

$$\max_{K,(i,\delta,z)} \Big\{ \Pr\big[x \xleftarrow{\$} \{0,1\}^n : \varphi^{in}(K,x) = (i,\delta,z) \;\big|\; x \in \mathsf{Dom}_{\mathsf{if}}(E1,K)\big] \Big\} \leq \varepsilon_{de(\varphi^{in})}.$$

$$\max_{x,(i,\delta,z)} \Big\{ \Pr\big[K \xleftarrow{\$} \{0,1\}^{\kappa} : \varphi^{in}(K,x) = (i,\delta,z) \;\big|\; x \in \mathsf{Dom}_{\mathsf{if}}(E1,K)\big] \Big\} \leq \varepsilon_{de(\varphi^{in})}.$$

By default, we assume $\varepsilon_{de(\varphi^{in})} = negl(n)$ is negligible.

Input functions in common inv-free blockciphers are indeed non-degenerate: e.g., key-prepended Feistel round has $\varphi^{in}(K,x) = \big(i,+,K\|\mathsf{right}_{n/2}(x)\|[0]_{n/2}\big)$ (see Eq. (4)) and $\varepsilon_{de(\varphi^{in})} = \max\{1/2^n, 1/2^{\kappa}\}$.

Theorem 2 (Differentiability of $E2^{\mathcal{P}}$). *Let $E2^{\mathcal{P}}$ be a blockcipher defined by Fig. 10 with keyspace $|\mathcal{K}| \geq \big(6(3|\mathcal{I}_{\leq n}|)^{\frac{1}{n}} + 5\big)|\mathcal{I}_{\leq n}| + 1 = O(poly(n))$. Assume that: (i) for all $j \in \{0,1,2\}$, the round $\Pi_j^{\mathcal{P}}$ is deterministic and $\varepsilon_{de(\Pi_j)}$-non-degenerate, and (ii) φ_1^{in} and φ_2^{in} are $\varepsilon_{de(\varphi_1^{in})}$- and $\varepsilon_{de(\varphi_2^{in})}$-non-degenerate respectively (see Definition 3). Then, when n is sufficiently large, there exists a differentiator $D2^{E,P}$ making $poly(n)$ queries and has advantage at least $1 - {m_{max}}^2/2^n - 2q^2\varepsilon_{de(\varphi_1^{in})} - 2q^2\varepsilon_{de(\varphi_2^{in})} - 6q^2/2^{m_{min}} = 1 - negl(n)$, where q is the number of* **IC***-queries made by $D2$ and S in total.*

We refer to [28] for its proof and Sect. 3.3 for the overview.

```
Algorithm $E2^{\mathcal{P}}(K,x)$
if $K \in \mathcal{K}^{(1)}$ then
  return $\Pi_3^{\mathcal{P}}\big(K\|\mathsf{kd}^{\mathcal{P}}(K),x\big)$
else        // $K \in \mathcal{K}^{(0)}$
  $u \leftarrow \Pi_1^{\mathcal{P}}(K,x)$
  return $\Pi_2^{\mathcal{P}}\big(K,u\big)$
end if

Algorithm $\Pi_j^{\mathcal{P}}(K,x)$ // $j \in \{0,1,...,5\}$
$(i_j,\delta_j,z_j) \leftarrow \varphi_j^{in}(K,x)$
$z'_j \leftarrow \mathcal{P}(i_j,\delta_j,z_j)$
$y \leftarrow \varphi_j^{out}(K,z'_j,x)$
return $y$

Algorithm $\mathsf{kd}^{\mathcal{P}}(K)$
$(i,\delta,z) \leftarrow f(K)$
$z' \leftarrow \mathcal{P}(i,\delta,z)$
return $z'$
```

```
Algorithm $(E2^{-1})^{\mathcal{P}}(K,y)$
if $K \in \mathcal{K}^{(1)}$ then
  return $(\Pi_3^{-1})^{\mathcal{P}}\big(K\|\mathsf{kd}^{\mathcal{P}}(K),y\big)$
else        // $K \in \mathcal{K}^{(0)}$
  $u \leftarrow (\Pi_2^{-1})^{\mathcal{P}}(K,y)$
  return $(\Pi_1^{-1})^{\mathcal{P}}\big(K,u\big)$
end if

Algorithm $(\Pi_j^{-1})^{\mathcal{P}}(K,y)$ //
$j \in \{0,1,...,5\}$
$(i_j,\delta_j,z_j) \leftarrow \gamma_j^{in}(K,y)$
$z'_j \leftarrow \mathcal{P}(i_j,\delta_j,z_j)$
$x \leftarrow \varphi_j^{out}(K,z'_j,y)$
return $x$
```

Fig. 11. Definition of the 2-call iterated blockcipher $E2^{\mathcal{P}}$.

7 Attack 3-Call Iterated Blockciphers

For 3-call iterated ciphers, $E3^{\mathcal{P}}(K,x)$ can take one of the following three flows:

- **Type-I**: $E3^{\mathcal{P}}(K,x) = \Pi_6^{\mathcal{P}}\big(K\|\mathsf{kd}_1^{\mathcal{P}}(K),x\big)$ for a 2-call KDF $\mathsf{kd}_1^{\mathcal{P}} : \{0,1\}^{\kappa} \to \{0,1\}^{2m_{max}}$ and a 1-call cipher $\Pi_6^{\mathcal{P}} : \{0,1\}^{\kappa+2m_{max}} \times \{0,1\}^n \to \{0,1\}^n$, or
- **Type-II**: $E3^{\mathcal{P}}(K,x) = \Pi_5^{\mathcal{P}}\big(K\|\mathsf{kd}_2^{\mathcal{P}}(K), \Pi_4^{\mathcal{P}}(K\|\mathsf{kd}_2^{\mathcal{P}}(K),x)\big)$ for a 1-call KDF $\mathsf{kd}_2^{\mathcal{P}} : \{0,1\}^{\kappa} \to \{0,1\}^{m_{max}}$ and two 1-call ciphers $\Pi_4^{\mathcal{P}}, \Pi_5^{\mathcal{P}} : \{0,1\}^{\kappa+m_{max}} \times \{0,1\}^n \to \{0,1\}^n$, or
- **Type-III**: $E3^{\mathcal{P}}(K,x) = \Pi_3^{\mathcal{P}}\big(K, \Pi_2^{\mathcal{P}}\big(K, \Pi_1^{\mathcal{P}}(K,x)\big)\big)$ for three 1-call ciphers $\Pi_1^{\mathcal{P}}, \Pi_2^{\mathcal{P}}, \Pi_3^{\mathcal{P}} : \{0,1\}^{\kappa} \times \{0,1\}^n \to \{0,1\}^n$.

The keyspace is partitioned $\mathcal{K} = \mathcal{K}^{(0)} \sqcup \mathcal{K}^{(1)} \sqcup \mathcal{K}^{(2)}$, such that $E3^{\mathcal{P}}(K,\cdot)$ follows **Type-I**, resp. **Type-II** encipherment if and only if $K \in \mathcal{K}^{(2)}$, resp. $K \in \mathcal{K}^{(1)}$. Formally, $E3^{\mathcal{P}}$ is defined in Fig. 12.

Theorem 3 (Differentiability of $E3^{\mathcal{P}}$). *Let $E3^{\mathcal{P}}$ be a blockcipher defined by Fig. 12 with keyspace $\{0,1\}^{\kappa}$, $\kappa \geq 2m_{max}\log_2|\mathcal{I}_{\leq n}| + 2m_{max}n + 6m_{max} + 4 = \Theta(poly(n))$. Assume that for $j = 1,2,3,4,5,6$, (i) the round $\Pi_j^{\mathcal{P}}$ is deterministic and $\varepsilon_{de(\Pi_j)}$-non-degenerate, and (ii) φ_j^{in} is $\varepsilon_{de(\varphi_j^{in})}$-non-degenerate (see Definition 3). Then, when n is sufficiently large, there exists a differentiator $D3^{E,P}$ making poly(n) queries to E and P and having advantage either $1/poly(n) - negl(n)$ or $1 - negl(n)$ for some poly(n) and negl(n) determined by φ_j^{in}, $j = 1,2,3,4,5,6$.*

We refer to [28] for its proof and Sect. 3.4 for the overview.

Acknowledgment. We thank an anonymous STOC '19 reviewer of [27] for suggesting considering lower bounds. We also thank Mridul Nandi, Wenfeng

```
**Algorithm** $E3^{\mathcal{P}}(K, x)$
**if** $K \in \mathcal{K}^{(2)}$ **then**
  **return** $\mathit{\Pi}_6^{\mathcal{P}}\big(K \| \mathsf{kd}_1^{\mathcal{P}}(K), x\big)$
**else if** $K \in \mathcal{K}^{(1)}$ **then**
  $s \leftarrow \mathsf{kd}_2^{\mathcal{P}}(K)$
  **return** $\mathit{\Pi}_5^{\mathcal{P}}\big(K \| s, \mathit{\Pi}_4^{\mathcal{P}}(K \| s, x)\big)$
**else**          // $K \in \mathcal{K}^{(0)}$
  $u \leftarrow \mathit{\Pi}_1^{\mathcal{P}}(K, x)$
  **return** $\mathit{\Pi}_3^{\mathcal{P}}\big(K, \mathit{\Pi}_2^{\mathcal{P}}(K, u)\big)$
**end if**

**Algorithm** $\mathsf{kd}_1^{\mathcal{P}}(K)$
$(i_1, \delta_1, z_1) \leftarrow f_{1,1}(K)$
$z_1' \leftarrow \mathcal{P}(i_1, \delta_1, z_1)$
$(i_2, \delta_2, z_2) \leftarrow f_{1,2}(K, z_1')$
$z_2' \leftarrow \mathcal{P}(i_2, \delta_2, z_2)$
**return** $z_1' \| z_2'$
```

```
**Algorithm** $(E3^{-1})^{\mathcal{P}}(K, y)$
**if** $K \in \mathcal{K}^{(2)}$ **then**
  **return** $(\mathit{\Pi}_6^{-1})^{\mathcal{P}}\big(K \| \mathsf{kd}_1^{\mathcal{P}}(K), y\big)$
**else if** $K \in \mathcal{K}^{(1)}$ **then**
  $s \leftarrow \mathsf{kd}_2^{\mathcal{P}}(K)$
  **return** $(\mathit{\Pi}_4^{-1})^{\mathcal{P}}\big(K \| s, (\mathit{\Pi}_5^{-1})^{\mathcal{P}}(K \| s, y)\big)$
**else**          // $K \in \mathcal{K}^{(0)}$
  $w \leftarrow (\mathit{\Pi}_3^{-1})^{\mathcal{P}}(K, y)$
  **return** $(\mathit{\Pi}_1^{-1})^{\mathcal{P}}\big(K, (\mathit{\Pi}_2^{-1})^{\mathcal{P}}(K, w)\big)$
**end if**

**Algorithm** $\mathsf{kd}_2^{\mathcal{P}}(K)$
$(i, \delta, z) \leftarrow f_{2,1}(K)$
$z' \leftarrow \mathcal{P}(i, \delta, z)$
**return** $z'$

// Definitions of $\mathit{\Pi}_j^{\mathcal{P}}(K, x)$ and
$(\mathit{\Pi}_j^{-1})^{\mathcal{P}}(K, y)$ are the same as Fig. 11
```

Fig. 12. Definition of the 3-call iterated blockcipher $E3^{\mathcal{P}}$.

Qi, Abishanka Saha, Sayantan Paul and Meiqin Wang and EUROCRYPT 2023 reviewers for fruitful comments. This work was partly supported by the National Natural Science Foundation of China (Grant No. 62002202, No. 61872359) and the Taishan Scholars Program (for Young Scientists) of Shandong.

References

1. Andreeva, E., Bogdanov, A., Dodis, Y., Mennink, B., Steinberger, J.P.: On the Indifferentiability of Key-Alternating Ciphers. In: CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 531–550. Springer (2013). https://doi.org/10.1007/978-3-642-40041-4_29
2. Avanzi, R.: A Salad of Block Ciphers. IACR Cryptol. ePrint Arch. p. 1171 (2016), http://eprint.iacr.org/2016/1171
3. Barak, B., Mahmoody-Ghidary, M.: Lower Bounds on Signatures From Symmetric Primitives. In: 48th FOCS. pp. 680–688. IEEE (2007). https://doi.org/10.1109/FOCS.2007.38
4. Barak, B., Mahmoody-Ghidary, M.: Merkle's Key Agreement Protocol is Optimal: An $O(n^2)$ Attack on Any Key Agreement from Random Oracles. J. Cryptology **30**(3), 699–734 (Jul 2017). https://doi.org/10.1007/s00145-016-9233-9
5. Barbosa, M., Farshim, P.: Indifferentiable Authenticated Encryption. In: CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 187–220. Springer (2018). https://doi.org/10.1007/978-3-319-96884-1_7
6. Bellare, M., Kohno, T.: A Theoretical Treatment of Related-Key Attacks: RKA-PRPs, RKA-PRFs, and Applications. In: EUROCRYPT 2003. pp. 491–506. LNCS, Springer (2003). https://doi.org/10.1007/3-540-39200-9_31
7. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the Indifferentiability of the Sponge Construction. In: EUROCRYPT 2008. pp. 181–197. LNCS, Springer (2008). https://doi.org/10.1007/978-3-540-78967-3_11

8. Biryukov, A., Khovratovich, D., Nikolic, I.: Distinguisher and Related-Key Attack on the Full AES-256. In: CRYPTO 2009. LNCS, vol. 5677, pp. 231–249. Springer (2009). https://doi.org/10.1007/978-3-642-03356-8_14
9. Black, J., Cochran, M., Shrimpton, T.: On the Impossibility of Highly-Efficient Blockcipher-Based Hash Functions. J. Cryptology **22**(3), 311–329 (Jul 2009). https://doi.org/10.1007/s00145-008-9030-1
10. Black, J., Rogaway, P., Shrimpton, T., Stam, M.: An Analysis of the Blockcipher-Based Hash Functions from PGV. J. Cryptology **23**(4), 519–545 (Oct 2010). https://doi.org/10.1007/s00145-010-9071-0
11. Bogdanov, A., Knudsen, L.R., Leander, G., Standaert, F.X., Steinberger, J.P., Tischhauser, E.: Key-Alternating Ciphers in a Provable Setting: Encryption Using a Small Number of Public Permutations - (Extended Abstract). In: EUROCRYPT 2012. pp. 45–62. LNCS, Springer (2012). https://doi.org/10.1007/978-3-642-29011-4_5
12. Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited. J. ACM **51**(4), 557–594 (2004). https://doi.org/10.1145/1008731.1008734, https://doi.org/10.1145/1008731.1008734
13. Carmer, B., Rosulek, M.: Linicrypt: A Model for Practical Cryptography. In: CRYPTO 2016, Part III. LNCS, vol. 9816, pp. 416–445. Springer (2016). https://doi.org/10.1007/978-3-662-53015-3_15
14. Choi, W., Lee, B., Lee, J.: Indifferentiability of Truncated Random Permutations. In: ASIACRYPT 2019, Part I. pp. 175–195. LNCS, Springer (2019). https://doi.org/10.1007/978-3-030-34578-5_7
15. Coron, J.S., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård Revisited: How to Construct a Hash Function. In: CRYPTO 2005. LNCS, vol. 3621, pp. 430–448. Springer (2005). https://doi.org/10.1007/11535218_26
16. Coron, J.S., Holenstein, T., Künzler, R., Patarin, J., Seurin, Y., Tessaro, S.: How to Build an Ideal Cipher: The Indifferentiability of the Feistel Construction. J. Cryptology **29**(1), 61–114 (Jan 2016). https://doi.org/10.1007/s00145-014-9189-6
17. Dachman-Soled, D., Katz, J., Thiruvengadam, A.: 10-Round Feistel is Indifferentiable from an Ideal Cipher. In: EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 649–678. Springer (2016). https://doi.org/10.1007/978-3-662-49896-5_23
18. Dai, Y., Seurin, Y., Steinberger, J.P., Thiruvengadam, A.: Indifferentiability of Iterated Even-Mansour Ciphers with Non-idealized Key-Schedules: Five Rounds Are Necessary and Sufficient. In: CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 524–555. Springer (2017). https://doi.org/10.1007/978-3-319-63697-9_18
19. Dai, Y., Steinberger, J.P.: Indifferentiability of 8-Round Feistel Networks. In: CRYPTO 2016, Part I. LNCS, vol. 9814, pp. 95–120. Springer (2016). https://doi.org/10.1007/978-3-662-53018-4_4
20. Demay, G., Gaži, P., Hirt, M., Maurer, U.: Resource-Restricted Indifferentiability. In: EUROCRYPT 2013. pp. 664–683. LNCS, Springer (2013). https://doi.org/10.1007/978-3-642-38348-9_39
21. Dodis, Y., Stam, M., Steinberger, J.P., Liu, T.: Indifferentiability of Confusion-Diffusion Networks. In: EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 679–704. Springer (2016). https://doi.org/10.1007/978-3-662-49896-5_24
22. Durak, F.B., Vaudenay, S.: Breaking the FF3 Format-Preserving Encryption Standard over Small Domains. In: CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 679–707. Springer (2017). https://doi.org/10.1007/978-3-319-63715-0_23
23. Even, S., Mansour, Y.: A Construction of a Cipher from a Single Pseudorandom Permutation. J. Cryptology **10**(3), 151–162 (Jun 1997). https://doi.org/10.1007/s001459900025

24. Füredi, Z., Simonovits, M.: The History of Degenerate (Bipartite) Extremal Graph Problems, pp. 169–264. Springer Berlin Heidelberg, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39286-3_7, https://doi.org/10.1007/978-3-642-39286-3_7
25. Gennaro, R., Gertner, Y., Katz, J., Trevisan, L.: Bounds on the Efficiency of Generic Cryptographic Constructions. SIAM J. Comput. **35**(1), 217–246 (2005). https://doi.org/10.1137/S0097539704443276, https://doi.org/10.1137/S0097539704443276
26. Grassi, L.: On Generalizations of the Lai-Massey Scheme: the Birth of Amaryllises. Cryptology ePrint Archive, Paper 2022/1245 (2022)
27. Guo, C., Lin, D.: Indifferentiability of 3-Round Even-Mansour with Random Oracle Key Derivation. Cryptology ePrint Archive, Report 2016/894 (2016)
28. Guo, C., Wang, L., Lin, D.: Impossibility of Indifferentiable Iterated Blockciphers from 3 or Less Primitive Calls. Cryptology ePrint Archive, Paper 2023/226 (2023), https://eprint.iacr.org/2023/226, https://eprint.iacr.org/2023/226
29. Hoang, V.T., Morris, B., Rogaway, P.: An Enciphering Scheme Based on a Card Shuffle. In: CRYPTO 2012. LNCS, vol. 7417, pp. 1–13. Springer (2012). https://doi.org/10.1007/978-3-642-32009-5_1
30. Hoang, V.T., Tessaro, S.: Key-Alternating Ciphers and Key-Length Extension: Exact Bounds and Multi-user Security. In: CRYPTO 2016, Part I. LNCS, vol. 9814, pp. 3–32. Springer (2016). https://doi.org/10.1007/978-3-662-53018-4_1
31. Holenstein, T., Sinha, M.: Constructing a Pseudorandom Generator Requires an Almost Linear Number of Calls. In: 53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012. pp. 698–707. IEEE Computer Society (2012). https://doi.org/10.1109/FOCS.2012.51, https://doi.org/10.1109/FOCS.2012.51
32. Hoory, S.: The Size of Bipartite Graphs with a Given Girth. J. Comb. Theory, Ser. B **86**(2), 215–220 (2002). https://doi.org/10.1006/jctb.2002.2123, https://doi.org/10.1006/jctb.2002.2123
33. Iwata, T., Kohno, T.: New Security Proofs for the 3GPP Confidentiality and Integrity Algorithms. In: FSE. pp. 427–445. LNCS, Springer (2004). https://doi.org/10.1007/978-3-540-25937-4_27
34. Kahn, J., Saks, M.E., Smyth, C.D.: A Dual Version of Reimer's Inequality and a Proof of Rudich's Conjecture. In: Proceedings of the 15th Annual IEEE Conference on Computational Complexity, Florence, Italy, July 4-7, 2000. pp. 98–103. IEEE Computer Society (2000). https://doi.org/10.1109/CCC.2000.856739, https://doi.org/10.1109/CCC.2000.856739
35. Kilian, J., Rogaway, P.: How to Protect DES Against Exhaustive Key Search (an Analysis of DESX). J. Cryptology **14**(1), 17–35 (Jan 2001). https://doi.org/10.1007/s001450010015
36. Kim, J.H., Simon, D.R., Tetali, P.: Limits on the Efficiency of One-Way Permutation-Based Hash Functions. In: 40th FOCS. pp. 535–542. IEEE (1999). https://doi.org/10.1109/SFFCS.1999.814627
37. Knudsen, L.R., Rijmen, V.: Known-Key Distinguishers for Some Block Ciphers. In: ASIACRYPT 2007. LNCS, vol. 4833, pp. 315–324. Springer (2007). https://doi.org/10.1007/978-3-540-76900-2_19
38. Kővári, P., T Sós, V., Turán, P.: On a problem of Zarankiewicz. In: Colloquium Mathematicum. vol. 3, pp. 50–57. Polska Akademia Nauk (1954)

39. Lampe, R., Seurin, Y.: How to Construct an Ideal Cipher from a Small Set of Public Permutations. In: ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 444–463. Springer (2013). https://doi.org/10.1007/978-3-642-42033-7_23
40. Luby, M., Rackoff, C.: How to Construct Pseudorandom Permutations from Pseudorandom Functions. SIAM J. Comput. **17**(2), 373–386 (1988). https://doi.org/10.1137/0217022, https://doi.org/10.1137/0217022
41. Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In: TCC 2004. LNCS, vol. 2951, pp. 21–39. Springer (2004). https://doi.org/10.1007/978-3-540-24638-1_2
42. Nandi, M.: On the Optimality of Non-Linear Computations of Length-Preserving Encryption Schemes. In: ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 113–133. Springer (2015). https://doi.org/10.1007/978-3-662-48800-3_5
43. Ristenpart, T., Shacham, H., Shrimpton, T.: Careful with Composition: Limitations of the Indifferentiability Framework. In: EUROCRYPT 2011. LNCS, vol. 6632, pp. 487–506. Springer (2011). https://doi.org/10.1007/978-3-642-20465-4_27
44. Rogaway, P., Steinberger, J.P.: Security/Efficiency Tradeoffs for Permutation-Based Hashing. In: EUROCRYPT 2008. pp. 220–236. LNCS, Springer (2008). https://doi.org/10.1007/978-3-540-78967-3_13
45. Rønjom, S., Bardeh, N.G., Helleseth, T.: Yoyo Tricks with AES. In: ASIACRYPT 2017, Part I. pp. 217–243. LNCS, Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_8
46. Rudich, S.: Limits on the provable consequences of one-way functions. Ph. D. Thesis, University of California (1988)
47. Shannon, C.: Communication theory of secrecy system. Bell Syst Tech J (28), 656–715 (1949)
48. Simon, D.R.: Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions? In: EUROCRYPT 1998. pp. 334–345. LNCS, Springer (1998). https://doi.org/10.1007/BFb0054137
49. Wagner, D.: The Boomerang Attack. In: FSE. pp. 156–170. LNCS, Springer (1999). https://doi.org/10.1007/3-540-48519-8_12
50. Wu, H., Preneel, B.: AEGIS: A Fast Authenticated Encryption Algorithm. In: SAC 2013. pp. 185–201. LNCS, Springer (2014). https://doi.org/10.1007/978-3-662-43414-7_10
51. Zhandry, M., Zhang, C.: Indifferentiability for Public Key Cryptosystems. In: CRYPTO 2020, Part I. pp. 63–93. LNCS, Springer (2020). https://doi.org/10.1007/978-3-030-56784-2_3